Consequence based design
An approach for integrating computational collaborative models (Integrated Dynamic Models) in the building design phase

Kristoffer Negendahl

PhD Thesis

Department of Civil Engineering
2016

DTU Civil Engineering Report R-329

DTU Civil Engineering
Department of Civil Engineering

Consequence based design

An approach for integrating computational collaborative models (Integrated Dynamic Models) in the building design phase

DTU Civil Engineering
Department of Civil Engineering

Innovationsfonden
FORSKNING, TEKNOLOGI & VÆKST I DANMARK
Grontmij SWECO

Consequence based design

An approach for integrating computational collaborative models (Integrated Dynamic Models) in the building design phase

15.01.2016

By Kristoffer Negendahl

Cover photo: Graph-based model modeled in Grasshopper™

Published by: Department of Civil Engineering, Brovej, Building 118, DK-2800 Kgs. Lyngby, Denmark

Request from: www.byg.dtu.dk

ISSN: 1601-2907

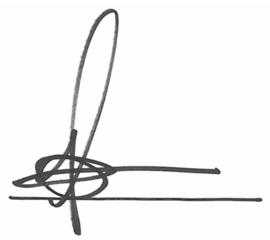
ISBN: 978-87-7877-419-4

Preface

This thesis concludes the Ph.D. work entitled: Consequence based design - An approach for integrating collaborative models (Integrated Dynamic Models) in the building design phase. The work was carried out between December 2011 and June 2015 between the Department of Civil Engineering, Technical University of Denmark (DTU) and Grontmij consulting engineers (now Sweco Engineering Consultant), Denmark. The project was financially supported by Innovationsfonden and Grontmij. The project concerns the use of integrated dynamic models in the aim to create High-Performance Buildings.

I would like to use this opportunity to thank my supervisors: Associate Professor and Head of Section, DTU Toke Rammer Nielsen, Head of Department, Grontmij Lars Hagstrøm, Founder and Partner Tredje Natur, Ole Schrøder, Architect Ph.D. Teresa Surzycka and my colleagues at DTU and Grontmij for rewarding discussions and exchange of ideas. Furthermore, I would like to thank 3XN, BIG, CCO, Kant, Lundgaard & Tranberg and Niels Bjørn for motivating and inspiring the work of this project
Kgs. Lyngby, Jan 2016

Kristoffer Negendahl



Til Sussi, Eva Bjørk og Anker

Content

Summary (US)

In the wake of uncompromising requirements on building performance and the current emphasis on building energy consumption and indoor environment, designing buildings has become an increasingly difficult task. However, building performance analyses, including those of building energy consumption and indoor environment, are generally conducted late in the design process. As a result, building performance evaluations are omitted in the early design where changes are least expensive. Consequence based design is a framework intended for the early design stage. It involves interdisciplinary expertise that secures validity and quality assurance with a simulationist while sustaining autonomous control of building design with the building designer. Consequence based design is defined by the specific use of integrated dynamic models. These models include the parametric capabilities of a visual programming tool, the building analyses features of a building performance simulation tool and the modelling and visualisation features of a design tool. The framework is established to enhance awareness of building performance in the early stages of building design, in the aim to create High-Performance Buildings. The project relies on various advancements in the area of integrated dynamic models. It also relies on the application and test of the approach in practice to evaluate the Consequence based design and the use of integrated dynamic models. As a result, the Consequence based design approach has been applied in five case studies. All case studies concern building design projects performed in collaboration with Grontmij and various Danish architectural studios. Different types of integrated dynamic models have been implemented and tested for the individual projects. The findings from each project were used to alter and define new ways to implement integrated dynamic models for the following project. In parallel, seven different developments of new methods, tools and algorithms have been performed to support the application of the approach. The developments concern: **Decision diagrams** – to clarify goals and the ability to visualize any relevant building performance. **AHP** – the use of Analytic Hierarchy Process to clarify differences between solutions on both qualitative and quantitative evaluations. **Termite** – the implementation of the BPS tool solver Be10 as a plugin for Grasshopper that enables live feedback of entire building energy consumption. **HQSS** – a quasi-steady-state BPS tool solver dedicated for fast thermal analyses in Grasshopper. **Moth** – an agent-based optimization algorithm implemented in Grasshopper that attempts to combine qualitative and quantitative evaluations during optimization. **Sentient models** – a method to listen to user behaviour in Grasshopper and decrease the space of solutions. **Surrogate models** – a test of machine learning methods to speed up any BPS feedback through surrogate models with Grasshopper.

This thesis demonstrates how integrated dynamic models may include building performance feedbacks, specifically feedbacks regarding energy consumption and indoor environment in the aim to create High-Performance Buildings. It further demonstrates the inclusion of quality defined performances un-associated with High-Performance Buildings. The thesis discusses ways integrated dynamic models affect the design process and collaboration between building designers and simulationists. Within the limits of applying the approach of Consequence based design to five case studies, followed by documentation based on interviews, surveys and project related documentations derived from internal reports and similar sources, this thesis can conclude that integrated dynamic models for these particular case studies can improve the speed of multiple and parallel performance evaluations, reduce working hours for the simulationists and are likely to improve the goal of creating High-Performance Buildings.

Resume (DK)

I kølvandet af kompromisløse krav til bygningers energiforbrug og indeklima er bygningsdesign en mere omfattende opgave end for bare få år siden. Analyser af bygningers energiforbrug og indeklima er generelt afgrænset til sent i designprocessen, hvorfor der sjældent tages hensyn til bygnings-performance i den tidlige designfase. Konsekvensbaseret design er en tilgang til den tidlige designfase, der involverer tværfaglig ekspertise. En tilgang der sikrer validitet og uafhængig kontrol af bygningsdesignet hos bygningsdesigneren. Konsekvensbaseret design er defineret ved det specifikke brug af integrerede dynamiske modeller. Disse modeller omfatter de parametriske muligheder forbundet med et visuelt programmeringsværktøj, bygnings-performance-analyser forbundet med et simuleringsværktøj samt modellerings- og visualiseringsfunktioner forbundet med et designværktøj. Den konsekvensbasserede designtilgang er etableret for at øge bevidstheden om bygningers ydeevne i de tidlige faser af bygningsdesignet, med det formål at skabe High-Performance Buildings. Projektet er afhængigt af forskellige typer af udvikling inden for integrerede dynamiske modeller. Dette indebærer en direkte anvendelse af modellerne og den konsekvensbasserede designtilgang i praksis. Som følge heraf, er den konsekvensbaserede designtilgang anvendt i fem casestudier. Alle casestudier er projekter udført i samarbejde med Grontmij og forskellige danske tegnestuer. Forskellige typer af integrerede dynamiske modeller er implementeret og testet for de enkelte projekter. Resultaterne fra hvert projekt blev brugt til at ændre og definere nye måder at implementere integrerede dynamiske modeller til anvendelse i de følgende projekter. I et parallelt forløb er syv forskellige udviklinger af nye metoder, værktøjer og algoritmer udført for at understøtte anvendelsen af tilgangen. Disse udviklinger inkluderer: **Decision diagrams** – for at afklare mål og visualisere relevant bygnings-performance. **AHP** – en anvendelse af Analytic Hierarchy Process for at klarlægge forskelle mellem løsninger på både kvalitative og kvantitative evalueringer. **Termite** - implementeringen af simuleringsværktøjet Be10, som et plugin til Grasshopper, der gør det muligt at forvalte direkte feedback fra bygningers energiforbrug. **HQSS** - et quasi-steady-state simuleringsværktøj dedikeret til hurtige termiske analyser i Grasshopper. **Moth** - en agent-baseret optimeringsalgoritme implementeret i Grasshopper, som forsøger at kombinere kvalitative og kvantitative evalueringer under optimeringen. **Sentient models** - en metode til at lytte til brugernes adfærd i Grasshopper og derved mindske løsningsrummet. **Surrogate models** - en test af machine-learning-metoder til at accelerere ethvert simuleringsværktøj vha. surrogat-modeller med Grasshopper.

Denne afhandling demonstrerer integrerede dynamiske modellers muligheder for at inkludere performance-basserede feedback, særligt behandles feedbacks med henblik på energiforbrug og indeklima med det formål, at skabe High-Performance Buildings. Afhandlingen demonstrerer yderligere mulighederne for at inkludere kvalitative bygningsperformances, normal uassocieret med High-Performance Buildings. Afhandlingen diskuterer måder integrerede dynamiske modeller påvirker design processen og samarbejdet mellem bygningsdesignere og simuleringsexperter. Ved dokumentation fra interviews, projektrelateret dokumentation fra interne kilder og spørgeskemaundersøgelser kan denne afhandling kan konkludere inden for rammerne af de testede casestudier, at konsekvensbasseret design har potentiale for at øge hastigheden af flere og parallelle performance undersøgelser, reducere arbejdsbyrden for simuleringsexperten og har potentiale for at forbedre muligheden for at skabe High-Performance Buildings

1. Introduction

The global resolution to increase sustainable development has resulted in growing pressure on building developers, building consultants and building designers to produce buildings that have a markedly higher level of environmental performance. Global initiatives such as the Kyoto protocol, agreements (e.g. the Paris agreement) and accords (e.g. the Copenhagen accord) through the United Nations Framework Convention on Climate Change (United Nations 1992) have marked a unified need of efforts to react upon global warming. These efforts are handled to various degrees on national and local levels and through private initiatives. In the EU, buildings are responsible for around 40% of the total energy consumption (EU, 2010). Which mean buildings are a main contributor to greenhouse gas emissions with 36% of EU's total CO_2 emissions (EU, 2010). By tightening building regulations and imposing various energy-saving measures, such as the encouragement of using sustainability certifications, the main idea is the average building energy consumption will be reduced. For instance in Denmark, building regulations alone have been estimated[1] (Danish Energy Agency, 2012) to reduce the energy consumption in new (residential) buildings from 105 kWh/m^2 in 1995 to 63.5 kWh/m^2 in 2010, and it is estimated[1] further reductions from 37 kWh/m^2 in 2015 to 20 kWh/m^2 in 2020. This means the environment and buildings are becoming increasingly sustainable, at least when it comes to energy usage. The growing pressure is being enforced by building regulations and clients who increasingly request the use of *green* design methods. This is also seen in the increased use of *green* certification systems. For instance BREEAM almost doubled the amount of certifications from 2008 to 2012 (8000 to nearly 16000 certified buildings) and in 2012 9669 sustainable certificates have been handed out to buildings in EU28 under the four leading schemes BREEAM, DGNB, LEED and HQE (Triple E et al., 2014).

In addition to the challenges of reducing global warming, a growing attention to human health in the built environment has similar effect on legislative and clients' requests for more *healthy* buildings. This is due to the fact that we as humans spend most of our time indoors. In the United States and European Union people spend 90% of their time either at work, at home or commuting between work and home (United States Environment Protection Agency, 1986). Controlled indoor environments are therefore directly coupled to human health and wellbeing. Nonetheless, building design today does not reflect the need for good indoor environments. It is estimated that nearly 30% of the modern building mass does not provide a healthy indoor environment (EPA, 1991).

---

[1] Estimations are based on the Danish building codes and calculated on 150 m2 single family house (Danish Energy Agency, 2012). These estimations may not represent the real building energy consumption, but will reflect the relative regulatory demands on the building energy use.

Unprecedented requirements for building performance

This pressure from legislation and demand from clients has led to significant changes in how buildings are being designed today and will be designed in the future, where more buildings are to be green and healthy. Although various experts have offered somewhat different interpretations of this, the consensus is that such green and healthy buildings must be characterized by a *measured high building performance* (Löhnert et al., 2003), over their entire life-cycle, in the following areas:

- Minimal use of non-renewable resources, (land, water, materials and fossil fuels)
- Minimal atmospheric emissions related to greenhouse gasses and acidification
- Minimal liquid effluents and solid wastes
- Minimal negative impacts on ecosystems
- Maximum quality of indoor environment, in the areas of air quality, thermal climate, illumination and acoustics/noise

These points can be condensed into two main objectives for *High-Performance Buildings* and *Integrated Design* (Löhnert et al., 2003):

- Minimal resource consumption over the entire building life-cycle – “green buildings”
- Maximum quality of indoor environment – “healthy buildings”

The current demand from authorities and building owners for High-Performance Buildings has resulted in unprecedented requirements with regard to energy consumption and indoor climate. To further complicate these demands, the two main of objectives of High-Performance Buildings are not in sync with one another, and often building designers need to find a trade-off between the two. The trade-off between a better global environment and a better built environment is not at all a simple task, which is why many researchers call for building performance evaluations from the early building design stage. Authorities and policy-makers are also aware of this problem, and many different European, national and local initiatives call for increasingly high levels of expertise in areas of building physics, materials and human physiology (Laustsen et al., 2011). The changing requirements and the level of documentation of these have amplified the pressure on the early design stage. Even though every intention to improve the building performance is directly connected to the goal of creating a more sustainable and healthy future, these stricter requirements pose many challenges. Therefore, the traditional design process driven by the architect, where engineers are included late in the design process is being challenged.

Integrated design

Several architectural studios and engineering companies today use different variants of integrated design methods. Many of these methods claim to implement the two main objectives of High-Performance Buildings. Some attempts to integrate the performance of building energy consumption and indoor environments have therefore been made both in practice and in theory (see 2.3.2). These attempts are based on more or less strictly prescribed methods, which often include an extensive use of building performance simulation tools and the assumption that clear performance objectives are the main drivers of the design process. As a result, the engineer (or engineering as a discipline) has in recent years gained more influence in this early design stage. While the reason is likely to be found in the increasingly more engineering-heavy tasks needed to comply with requirements for indoor climate and energy consumption as mentioned above, what remains to be seen today is a general adaption and use of such methods by *those who design buildings*[2] (de Souza, 2009).

There is unfortunately still a large gap between those who simulate and those who design (Mahdavi 1998). Mahdavi follows the gap back to the industrial revolution where the division between these professions grew in magnitude with the advancements in the field of sciences and materials. Both groups separately evolved within a mono-disciplinary environment and catered their services within a linear and fragmented building industry. Still today the gap is dividing the discipline of “crafting buildings” in two, and still today the challenge to collaborate is prevailing. An efficient solution to close the gap remains as of today unsolved. The prior efforts in integrated design tend to favor engineering aspects over aesthetics, functionality, accessibility, etc. One may only speculate if shortcomings of these methods have caused damage to the overall collaboration between the building designer and the simulationist during the early design stages. It would be beneficial for the design process to develop tools and methods that show more clearly the consequences of the performance criteria while it includes the architectural qualities as equal and important factors. In this way High-Performance Buildings may even be designed with more care than those that define “green and healthy” buildings.

1.1 The framework for Consequence based design

To evaluate certain performance requirements the design team needs to measure and carefully analyze the consequences of their design choices related to these requirements. While there are several ways to evaluate building performance, the utilization of dedicated software programs - also called building performance simulation (BPS) tools - provides one of the fastest and most flexible means of performance evaluation in the early design stage.
According to Souza (2012) it is paramount to embrace and use the division of interests and the separated world views of the building designer and the

[2] To avoid a categorization of the members in the design team based on their academic background, this thesis will follow Souza’s (de Souza, 2009) terminology: building designers are those who design buildings and simulationists are those who process and analyze data from simulation and calculation tools.

simulationist. To do this with a tool, in a way that contributes for the best possible outcome of creating High–Performance Buildings, the tool has to have to include the options of:

1. Selective and changing objectives
2. Easy to comprehend, manageable proportions of input and output
3. Visual impactful and fast response to specific parameters and/or set of conditions
4. Options to simplify and short cut the achievement of a desired state

Such tools do not exist off the shelf. As most BPS tools are *designed by engineers to accommodate engineers* (Klitgaard et al., 2006) little support of Souza's four points is implemented in BPS tools today. Also, no BPS tools are built to consolidate interdisciplinary collaboration. But many tools may be used in a collaborative process or as a part of a collaborative automated system. Many attempts to mend the disconnection between the BPS tools and the building designers have been made. These existing attempts ranges over systems, methods and tools are thoroughly discussed in 2.4.
One way of getting closer to a BPS tool with respect to separated world views of the building designer and the simulationist is to implement the design criteria and objectives in an integrated dynamic model (Negendahl, 2015a).

"An integrated dynamic model is basically a combined model composed of a geometric model controlled in a design tool (CAD) dynamically coupled to a visual programming language (VPL) which is again coupled to a building performance simulation (BPS) tool." (Negendahl, 2015a)

Integrated dynamic models can be considered as an implementation of a collaborative system of tools, where the BPS tools are a central part of the performance evaluations in a system that evaluates the "green" and the "healthy" elements of building design proposals. The VPLs are the backbone of parametric design methods. Considering that parametric design methods have become a natural part of the early design stages for many architectural studios, the integrated dynamic models may be easier to implement as opposed to the use of dedicated BPS tool environments and other alternative methods. In many ways, the inclusion of a VPL amplifies the dynamics of the design process as the parametric models are far more dynamic, open and flexible than fixed models. Parametric modelling is based on consistent relationships between objects, thus allowing changes in a single element to propagate throughout the system of tools. The inclusion of BPS eliminates the idea of guessing how a building performs. This is done by continuously simulating the performance of design changes while visualizing the results in the design tool. Therefore, Consequence based design is basically defined as a proposal to apply integrated dynamic models from the early design process, as an answer to Souza's (2012) four criteria, and to achieve High-Performance Buildings. This approach differs from many other similar performance-based approaches by the explicit use of the combination of design tools, VPLs and BPS tools thus being an explicit model-centric approach. How

Consequence based design approach separates itself further from the integrated design method is discussed in section 1.5.

The integrated dynamic model

The general concept of coupling the design tool with a BPS tool is not a novel idea, nor is the method in which these tools are coupled; through a programming language. Many researchers[3] have explored to use middleware between the design tool and BPS e.g. Bazjanac (2004) through IFC (BuildingSMART, 2013) and more recently dedicated software like ZEBO (Attia, 2012) and Virtual Design Studio (VDS) (Michael Pelken et al., 2013; Zhang et al., 2013). Holzer et al. (2009) and Sanguinetti et al. (2010) was some of the first recorded attempts in combining the design tool and the BPS tool with a scripting tool (Holzer et al. used Generative-Components, Sanguinetti used RhinoScript). None of these researchers however envisioned the method to be utilized as an early design stage analysis platform or generic model environment where the models themselves where independent/unbound of specific BPS tools and the model could span over multiple analyses areas. At the time where this project was defined dynamically coupling BPS tools and design tools over VPLs where uncommon practice in the building industry and few researchers has studied this phenomenon. At this time no present definition of such a model existed. The integrated dynamic model is basically the authors own definition of such a model.

---

[3] A detailed analysis on the existing coupling methods between the design tool and BPS tool is found in 2.4

1.2 Problem description

Designing High-Performance Buildings involves elements of expertise deriving from multiple professions such as architects, civil, mechanical and electrical engineers. It draws its resources from many diverse disciplines including physics, mathematics, material science and human behavior (Malkawi, 2004). With current emphasis on sustainability, including building energy consumption and indoor environment, design requirements from the involved disciplines have become more important in the early design stages. As a consequence building performance simulation (BPS) is increasingly used to support the design of buildings.

Numerous tools are used in a multitude of ways to predict building performance. This thesis focuses on computer-based models for energy and indoor environment analyses in the early design stage. Even though BPS tools are used to support building design, building performance simulations in the early design stage are still limited in practice.

The tools available today are far from ideal to support the early design stage. Malkawi (2004) argues that the existing BPS tools in many cases are limited to the final design stage. Klitgaard et al. (2006) even argues that most BPS tools are developed to accommodate *verification* of building design rather than the *exploration* of building design. Building designers still seem to prefer creating and exploring design options in dedicated design tools. Tools that support the concept of a sketch and the freedoms associated with design tools such as (ArchiCad, Sketchup, Revit, Rhino, Maya, etc.) (Hermund, 2012). Building design is thus created in design tools that do not provide building performance feedbacks. As a result the most prevalent method of receiving performance feedback in the early design stages is associated with either manually (re)modeling the designs in dedicated BPS tools or with a manual import and export task of the geometry. The introduction of building information modeling, BIM (specifically referring to the gbXML (gbXML.org, 2014) and IFC (BuildingSMART, 2013) standards), sought improvement, when teams are working with separated models. These formats have (yet) not succeeded in solving the many interoperability challenges between design tools and BPS tools, in a way that makes performance evaluations in design tools feasible (Toth et al., 2012). Even though the concept of a common reference model makes sense in all stages of building design, the early stages are often detached from any form of building information model. Seen from a technical point of view this is mainly due to the fact that many of the tools (both design tools and BPS tools) are yet to implement resilient tool integration.

Acknowledging that requirements for building design consist of *quantitative elements* (i.e. yearly consumed energy, amount of daylight, cost etc.) and *qualitative elements* (i.e. social impact, spatial planning, aesthetics, etc.), building design aims to satisfy multiple criteria in addition to measurable performances. This implies that building design is evidently connected to role-definitions and collaborative processes, and it also implies that the utilization of building performance has to respect the broad extent of both quantitative and qualitative elements of building design.

Many researchers have addressed the role-definitions and the process of building design in the aim to improve building performance. Hensen (2002), when focusing on the quality assurances of building performance simulations, among other arguments states the inclusion of specialist knowledge in the early design stage as paramount to achieve a valid ground for informed design. This thesis is built upon the assumption that buildings are designed and built by design teams (Löhnert et al., 2003) deriving from multiple professions such as architects, civil, mechanical and electrical engineers. Thus, it is natural to assume that different types of members and different interests exist within the design team. The key of this formulation is; not all projects are built upon a shared agreement that High-Performance Building design is the main objective.

One of the later attempts in reaching out and influencing the design process is Petersen (2011) who focused on affecting design decisions made in the early design process by introducing parametrical functionalities in a BPS tool (iDbuild (Petersen and Hviid, 2012)). This lead to faster performance feedback that could be applied as design changes in the designers' model. However, Petersen (2011) endorsed the need for further research in ways to influence the building designer acceptance of *the interdisciplinary, integrated design process as the general work form*. This includes better ways to integrate simulation-based design support as a useful input to the actual geometry of the given building. In other words, to further integrate the building designer and the building designer's geometrical representations of the building into the process of evaluation and optimization are needed.

Qualitative and quantitative requirements are handled very differently in building design. Most quantitative requirements, such as building performance requirements directed from legislation, are calculable. Given the right BPS tool, and given sufficient data available in the early design stage, a performance metric may help the design team to improve the building design. The qualitative requirements may not have any targetable metric and are in many cases purely based on human evaluations. This may be expressed as a need for new developments in methods, tools, algorithms and models that can sufficiently support qualitative evaluations of project specific design concepts. Such developments would allow the design team to make qualitative judgments in the exploration of architectural expressions and concepts while receiving building performance support in both the qualitative and the quantitative aspects of building design. In conclusion there is a need for tools and robust methods for performance support in both the qualitative and the quantitative aspects of building design.

1.3 Motivation for the thesis

Essential steps towards an innovative, integrated design and operation environment are needed to be successful in reaching the intents of High-Performance Buildings in practice. The ultimate goal illustrated in Figure 1 has been defined by Hensen (2004) to:

"provide tools, knowledge and procedures for integrated design and operation processes which lead to innovative, elegant and simple building designs with (a) a balanced attention to the value systems of the building occupier, building owner and the environment, (b) a better quality, (c) a shorter design time, and (d) lower life-cycle costs."

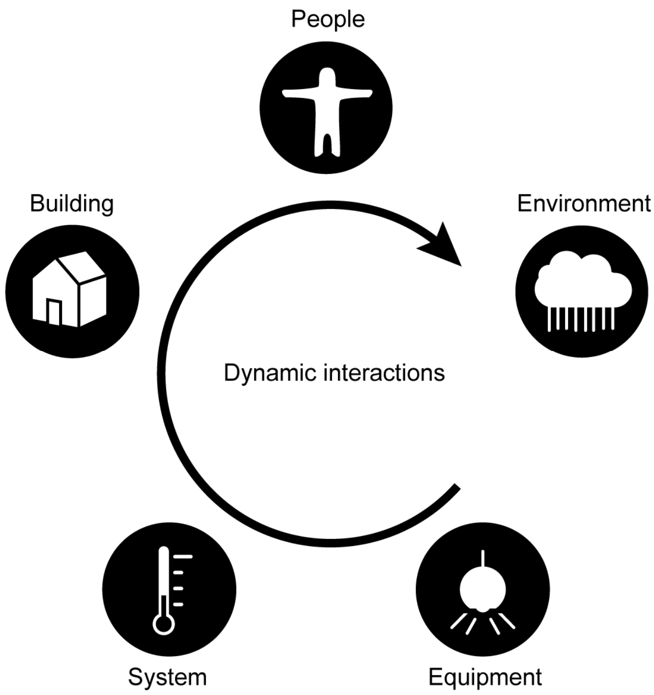


Figure 1 Dynamic interacting sub-systems in a building context inspired by (Kiesler, 1939) and (Hensen, 2004)

1.4 Aims and objectives

The project seeks to introduce Consequence based design as a new approach to collaboration within the design team. The aim is to increase the likelihood of creating High-Performance Buildings by the application of the Consequence based design approach to real building design projects (case studies) with various practitioners in the Danish building industry. The Consequence based design approach is defined by *the explicit creation and utilization of integrated dynamic models in the early design stage*, where the integrated dynamic model is defined as:

An Integrated dynamic model refer to a type of digital working model that links a selection of building performance simulation (BPS) tools to a design tool user interface through a visual programming language (VPL). Integrated dynamic models are meant to dynamically combine the typical architectural working models (geometry models) and the typical separate engineering mode (simulation models) in a distributed, linked model environment.

Two overall aims concerning Consequence based design were formulated.

The primary aim of the project is to envision, implement and document the use of integrated dynamic models.

This includes: The way integrated dynamic models affect building performance, specifically energy consumption and indoor environment. And the way integrated dynamic models affect the design process and collaboration between building designers and simulationists

The secondary aim is to envision, implement and document optimization processes coupled with integrated dynamic models.

Such optimization processes will rely on single- or multi-objective optimization algorithms e.g. evolutionary algorithms (Zitzler et al., 2001) and agent based models (Barbati et al., 2012), and they will draw on experience from previous methods (e.g. (Nielsen, 2002; Pedersen, 2006)) to apply optimization of building performance.

1.5 Scientific goals

The main goal of this project is to improve the technical foundation facilitating the Integrated Design method (Löhnert et al., 2003), thereby increasing the likelihood of achieving High-Performance Buildings. This means the main scientific goal is to pursue the increased use of building performance evaluation methods within the early design stages. The Consequence based design approach is meant to separate itself from the original Integrated Design method (and other dedicated performance-driven design approaches, which main intent is to improve or optimize selected performance based objectives) in few places. It differs mainly from performance-based design method as defined by Kalay 1999 by acknowledging that performance objectives are not necessarily absolute, relative to one another. Kalay argues objectives (behaviors towards a certain satisfaction function) that one “must first assign to each of [objective] a relative weight” (Kalay, 1999). To put a weight on any objective function, one must be aware of the function in relation to every other possible objective function. Instead the Consequence based design approach enables the building designer to evaluate a particular solution in either or both qualitative and quantitative criteria without inferring which performance metric is an objective. This is necessary to achieve High-Performing building design without compromising the comprehensive architectural expressions and concepts. The approach is basically to show the consequence of design choices of performance to the building designer through integrated dynamic models. In this way the project seeks to formulate an approach to the integrated design process that emphasizes the consequence of any design choice made during the early design stage.

1.6 Development goals

The goal is to build the approach on prior work of integrated design methods developed and used at the department (DTU Civil Engineering). The goal is to develop ways to use integrated dynamic models that may increase the likelihood of creating High-Performance Buildings. This includes the goal to investigate various boundary stages of the integrated dynamic model and to develop algorithms, tools and methods that may improve integrated dynamic models.

The applications (implementations) of integrated dynamic models will be a future direction in teaching and learning and will be a good starting point for one or several new courses teaching the ways to manage and analyze multiple and parallel objectives in building design. The use of programming languages and integration of multidisciplinary design goals will also accentuate the ongoing development of Building Information Modeling (BIM). By transposing to more data-related 3d-models in the early design stage the potential of transition to later BIM integration is likely to increase.

The project will benefit Grontmij by defining future ways of collaboration with their partners which will help the company to be competitive in the short and long term perspective. Integrated dynamic models will give the company experience with advanced optimization methods early in the design stage and the opportunity to win larger prestige projects, since the approach seeks to improve and accommodate innovative architectural concepts. The project will seek to improve the precision of engineering consulting and technical base of communication with the benefits derived from integrated dynamic models and the Consequence based design approach.

1.7 Commercial targets

The opportunities to develop tools that couple existing design tools and different BPS tools during the course of the project have direct commercial value. And there are opportunities to develop new tools, algorithms and methods to support Consequence based design. However, the approach and thus all relevant codes developed during the project will be open to Grontmij, their cooperative partners and collaborating parts of DTU. Thus, all development and programmed prospects are regarded as an indirect commercial value. The level of competence within the area of integrated dynamic models and Consequence based design is estimated as a vital value for the company. Grontmij is enforcing their own strategies towards a more integrated future so this project can become a central part of their expectations. In the prospect of teaching students at DTU or other institutes, the development of the integrated dynamic models will strengthen the knowledge within dynamic couplings, development of algorithms, tools and methods, thereby opening other commercial perspectives directly related to this project.

This Ph.D. project has been carried out as an industrial Ph.D., which means the project has great potential to be applied in practice. The main investor *Innovationsfonden* (2015) argues that once the project is carried out at both the university and the company, the research project must offer the company the possibility of *solving specific research and development tasks that create growth and employment*. Additionally, the research should strengthen the relationship between the business sector and the university to advance new research. Between these lines, there are several potential interest conflicts. Some are obvious, such as new developments formed during the project that might be of great interest for industry at large. The company might attempt to keep such developments to itself to avoid competition. Other interests are imposed by third parties and are more difficult to foresee.

1.8 Research methods

Methodology

As this research project per definition is an industrial Ph.D-project the opportunities to apply the research in "real life" is very high. Unlike traditional Ph.Ds industrial Ph.Ds has (an emphasized) direct contact with practice and thus theoretically able to weave practice and research together. This means research aims (Section 1.4) can be applied to practice. The prospect is to obtain immediate feedback from practitioners of different disciplines in the matter of integrated dynamic modeling and the application of Consequence based design. This instigates that the research has to be applied in a practice-based research form, partly by the means of practice and to advance knowledge within practice and finally to gain new knowledge regarding the use of integrated dynamic models. Frayling (1993) describes how research applies to the realm of design. This project fits into two categories of research, the first; *"into art and design",* which is the research on the theoretical approaches on building design and in this case building performance evaluations, and optimization. Secondly this project falls into the category of *"research through art and design"* as Frayling (1993) describes which may apply to projects that customizes *a piece of technology, to do something no one has done before, and communicating the results*. This definition can directly be applied to the integrated dynamic models and the software and tool developments this project has delivered. In essence this project can be defined as case-study based research project told through self-reflective ethnography, supported by surveys and interviews. Basically this Ph.D-project is built on pushing forward integrated dynamic models in architectural competition projects (Case studies – Section 4.1). The author is part of this process and therefore a stakeholder, an actor and an observer in each case study. Similar Ph.d-projects in Denmark built around this model can be mentioned are (Banke, 2013; Sattrup, 2012). The main challenge of this research method is to apply the practice-based research in an objective and unbiased way and to test research questions under circumstances that can be established for general purposes representing the larger industry. The limitations of this research method are the relative narrow amount of empirical data which is possible to generate during the course of the 3-year project. Every case study has limited amount participants who are able to give consistent feedback. And each case study is likely to be different from project to project increasing risk of inconsistent comparisons between projects.

Observations of the design team participants are chosen to construct and quantify relevant research questions, and where it is possible consistent questions and formulations are used across case studies. More the details of the surveys are found in section 4.1. Three of the central research questions given to the design team are:

- Have the interactions between members of the design team been affected by the use of integrated dynamic models in this project? If so, how?
- Has the use of integrated dynamic models had an impact on the simulated performance such as energy consumption, indoor environment and/or sustainability of this project? If so, how?
- Has the use of integrated dynamic models had an impact on other decisions in this project? If so, which?

The intention of surveying each case study is to generate a sufficient base of documented feedback to drive more general conclusions on Consequence based design. Where the surveys are unable to provide sufficient feedback further interviews and citations from internal documents will be needed. This research method is likely to create varying quality of documentation. Therefore conclusions may be highly coupled with individual case studies and results may be difficult to apply across practice. To test tendencies and findings on Consequence based design outside the bounds of (somewhat) controlled case studies surveys will be collected through internet survey forms unassociated with the case studies. This research method will not ensure absolute conjecture but is likely to either show similarities or differences of research questions relevant to substantiate and demonstrate the two main objectives of this Ph.D. project (Section 1.4)

In parallel to the applied research with practice-based case studies, research projects ministered by the author and students under the authors supervision has been driven the Ph.D-project in directions where practice-based projects could not. These projects reflected parts of ideal realities in a closed controlled environment useful for developing new tools, algorithms and methods. Some of these developments are described in Section 4.4 and some have been applied in practice (Case studies – Section 4.1).

Worth acknowledging is that these research methods constructed to support and examine the research aims only concerns a small part of overall organizational change with the particular focus on tools and models. Nonetheless successful organizational change is normally the product of people, processes and tools.

The research methods used throughout this Ph.D are described below in roughly the order in which they were used in the research.

Literature review

A literature review was conducted throughout the entire project. This raised the insight of the state of the art of BPS tools, design tools, as well as design and optimization methods used in the early design stage. Subjects that have been extensively reviewed are: performance evaluation methods, BPS tool and coupling methods, integrated design, Performance Based Design, optimization methods and methods to evaluate quality defined criteria. In addition to the reviews written as background for the various papers (Negendahl and Nielsen, 2015; Negendahl, 2014a, 2014b; Negendahl et al., 2015), a detailed review article was published in Automation in Construction (Negendahl, 2015a).

Interviews and observations

Interviews were conducted with internationally recognized practitioners. The aim was to learn about the design development tasks in practice. The interviews were informal and applied during specific project developments and "on the sideline[4]" on topics of general design methodology concerning parametric design and engineering. The interviews revolved around requirements for methods, tools and models to successfully support the early design stage.
Observations were conducted on the design/simulation activities of students when developing integrated dynamic models. The aim was to obtain empirical data about the elements used to structure the inclusion of High-Performance design criteria with the early stage design process.

Case studies

During the course of the project the author has participated in 12 different "real-life projects" in collaboration with Grontmij and various architectural studios. Five of these projects have been subject to the approach of Consequence based design or subject to investigations in relation to applied integrated dynamic models. These five case studies is discussed in detail in chapter 4.

Surveys

Seven surveys have been performed during the course of the project. The surveys have been an attempt to quantify the qualitative research. Five of the surveys will only provide case specific feedback, and two is meant to gather information from a broader audience. The questions are carefully produced in the

[4] Various interviews were performed during external stays in New York and London. The interviews were primarily focused on architectural studios; however a few consultants and other researchers in the field have provided valuable knowledge to the project.

same manner across the case studies, why general conclusions may be taken across the case studies. All surveys revolved around the use of integrated dynamic models and the character of Consequence based design approach in the design process. Some of the surveys were performed in relation to case studies and others were performed in open forums of building designers and simulationists. The surveys include the current state of integrated dynamic models, how practitioners and researchers use them, and how they want to use them in the future. To assure the highest level of conformity these surveys were conducted with same phrasing and language and based on the evaluation method in EN15251 annex H (BSI, 15251, 2006).

Teaching and supervision

13 student projects (primarily master students) have been exposed to the ideas of Consequence based design. The aim of the student projects was to give insight in the design development of High-Performance Buildings and to develop and apply the approach of Consequence based design and integrated dynamic models in controlled design spaces (as opposed to the practice-based case studies).

Modeling methods and developments

The analytic focus in this project concerns; building energy, indoor climate and sustainability. Therefore these same subjects are the main focus of the integrated dynamic models. This again means that the integrated dynamic models have been linked through different external tools relevant for the analysis of building energy, indoor climate and sustainability. The method used in this project have focused on the coupling of existing BPS tools commonly used in engineering practice e.g. Be10 (SBI, 2013) and Energy+ (U.S. Department of Energy, 2013). In the same time it is natural to expect building designers prefer to use existing design tools in the early design stage. The models therefore accommodates existing design tools such as Rhinoceros3D (Rhino) (Robert McNeel & Associates, 2013a) commonly used by building designers. The design tool and the BPS tool are linked by means of programming languages such as a VPL like Grasshopper3D (Grasshopper) (Robert McNeel & Associates, 2013b), or lower level languages such as Python or Java.

Optimization of High-Performance Building related criteria is implemented into the models as it has been done in previous objective based search methods using optimization algorithms e.g. (Nielsen, 2002), (Pedersen, 2006). The Consequence based design approach departs itself from these previous methods as it attempts to include and obtain performance criteria reminiscent of “architectural qualities” and it seek to respect architectural and intellectual subjectivity within optimization. Regarding design relations and design choices of “qualitative performance”, one must notice the distinction between the two notions, A. & B.

A. Design relations can be defined by underlying natural constraints such as Le Corbusier's derived Fibonacci curves (Lidwell et al., 2003). As a result, some specific design criteria might be applicable as mathematical constraints. Such constraints can be translated to various boundary conditions in optimization processes in e.g. Multi-objective optimization.

B. Design choice is defined as the qualitative assessments of design, which include aesthetics, style, firmness, usefulness and delight (Brooks, 2010). The design choice shows the importance of subjectivity and human opinion.

This approach seeks to include qualitative performance as boundaries within optimization models by the means of human assessment – notion B. Since optimization processes have difficulties to handle subjective opinions (Mora et al., 2008), the project had to accommodate the investigation and formulation methods of how to implement subjectivity in optimization.
The method in which integrated dynamic models have attempted to include subjective (or qualitative) performance have been handled as described by Duarte (2001). Here optimization is used to find non-dominating solutions in an unknown solution space based on quantitative performances in which only the "best performing solutions" are presented. Subsequently, it is up to the architect (or design team) to pick and choose subjectively among the solutions that may have qualitative properties. Any model that utilizes (single or multi-objective) optimization relies on existing algorithms, tools and theory on optimization.

Various software developments were conducted to advance simulations for performance predictions of representative case studies. Other developments were conducted to improve the coupling mechanisms between design tools and BPS tools. And yet other developments were performed in areas of optimization algorithms and machine learning algorithms. The process of these developments occurred over several iterative changes, and was based on the four step implementation procedure: 1) specification, 2) implementation, 3) verification and 4) testing. The specification is the required functionality of the development. The implementation allows the functions to be executed in a working computational environment. Verification ensures that the developments' subroutines work as specified. Testing relates the development to design of High-Performance Buildings in theory and practice.

1.9 Thesis outline

This thesis presents the Consequence based design approach by applying integrated dynamic models in theory and practice. The thesis is structured into three chapters:

· 2. Building design and performance
· 3. Optimization
· 4. Consequence based design

The two first chapters give an introduction, a review and a conclusion. These two chapters reflect the background needed to investigate the: *envisioning, implementation and documentation of the use of integrated dynamic models* and *envisioning, implementation and documentation of the use of optimization processes coupled with integrated dynamic models* following the two main objectives in section 1.4. The third chapter “Consequence based design” is divided into two main sections:

· 4.1 Case studies - the application of integrated dynamic models
· 4.4 Developments - to support Consequence based design

These two sections reflect the two types of research developments generated in this project – case studies performed with Grontmij and their various partners, and the developments of new tools, new algorithms and new methods. To properly organize the structure of these developments, it should be noted that some developments and case studies have been overlapping; for example, the tool “Termite” (see section 4.4.3) was used in some of the case studies. Therefore the whole project is divided into four phases as shown in Figure 2. The idea behind this structure is to distinguish the evaluation of integrated dynamic models that can be applied today and the development that is needed to support the Consequence based design approach in the future.

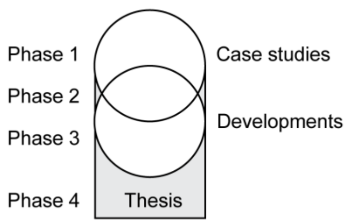


Figure 2 The structure of the project “Consequence based design” (Chapter 4). Phases 1-4: case studies cover various projects performed with Grontmij and different architectural studios. Developments cover individual development projects of new methods, tools and algorithms established by the author and in collaboration with students and other researchers.

In Figure 3 the method behind the development of the Consequence based design approach is illustrated. Once requirements of integrated dynamic models and Consequence based design have been established, the models are “put into practice" in case study projects (phases 1 & 2). The feedback from each project generates a new and improved knowledge base, which is used to define new requirements for new integrated dynamic models and the Consequence based design approach. In parallel (phases 2 & 3) as the knowledge base develops various needs for improvements that are difficult to apply in practice new developments are put into theory. With the aid of master students and other researchers in the field, different initiatives are taken to improve integrated dynamic models. Some of these initiatives focus on the process to collaborate within the design team; other initiatives are more technical such as the development of optimization algorithms and new BPS tools.

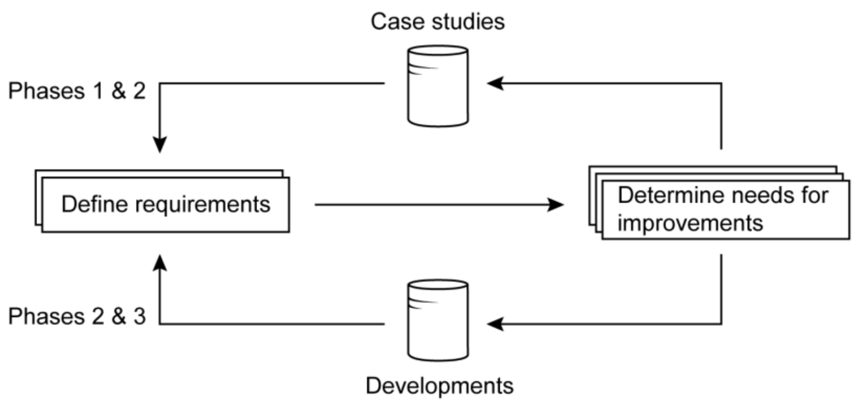


Figure 3 The method of development of "Consequence based design". Phases 1&2 utilized case studies (real case studies performed with various architectural studios) to define requirements of integrated dynamic models and to determine the needs for improvements of these models. Phases 2&3 applied the same procedure, only the case studies are substituted with developments of tools, algorithms and methods.

Chapter 2 addresses the need for better and faster evaluation of building performance in the early design stage. It lists the motivations for design teams to use integrated dynamic models and the approach of Consequence based design. And finally it covers a review of how BPS tools are used and how they can be integrated into the early design stage.

Chapter 3 reports why optimization is rarely used in the early design stage. This is done by reviewing current developments in the field and discussing the type of procedures needed to apply optimization algorithms in integrated dynamic models.

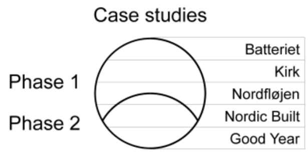


Figure 4 Some, but not all of the case studies have been combined with the new developments of tools, algorithms and methods.

Chapter 4 first presents five case studies where integrated dynamic models have been used in practice (
Figure 4). Each case study consists of an introduction, method, results, and conclusion. Secondly, an overview and collection of findings from the case studies are presented. Then other practitioners' perspectives are presented along with a discussion on a survey on the use of integrated dynamic models. After this, the new developments of tools, methods and algorithms are presented. Seven different Developments are presented; each project consists of an introduction, method, results, and conclusion. Some projects require additional background information, which is not found in Chapter 2 and 3; therefore, some extra references have been made here.

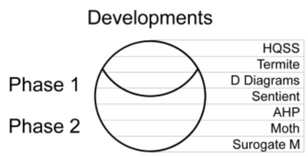


Figure 5 The Developments cover seven individual developments. Some are of more technical nature than others.

Chapter 4 concludes with some final remarks on integrated dynamic models and the general approach of Consequence based design.

Chapter 5 gives a conclusion and suggestions of future developments in the area of integrated dynamic models and Consequence based design.

Five articles and one poster are appended to the thesis and found in Appendix A.1-A.6. These publications document additional findings relating to the approach of Consequence based design. The publications either relate to specific case studies (Section 4.1) and/or specific developments (Section 4.4). Their relation to the case studies and developments is clarified in the relevant sections.

1.10 Publications

The contributions to journals and conferences written as part of this Ph.D.-study are listed below. The publications are found in Appendix A

Articles in journals

K. Negendahl (2015) Building Performance Simulation in the early design stage: An introduction to Integrated Dynamic Models, *Automation in Construction*

Negendahl, K., Nielsen, T.R. (2015) Building energy optimization in the early design stages: a simplified method, *Energy and Buildings*

Contributions to conferences

Negendahl, K., Perkov, T., Heller, A., (2014) Approaching Sentient Building Performance Simulation Systems, *Proceedings of eCAADe 2014,* presented by K. Negendahl at eCAADe in Newcastle, UK

Negendahl, K., (2014) Parametric design and analysis framework with integrated dynamic models, *Proceedings of the 3rd International Workshop on Design in Civil and Environmental Engineering (DCEE3),* presented by K. Negendahl at DCEE3 in Kgs. Lyngby, DK

Negendahl K., (2014) Parametric City Scale Energy Modeling Perspectives on using Termite in city scaled models, *iiESI European Workshop*, poster presentation, presented by K. Negendahl at iiESI in Copenhagen, DK

Negendahl, K., Perkov, T., Kolarik, J., (2015) Agent-based decision control - how to appreciate multivariate optimisation in architecture, *5th Design Modelling Symposium - Modelling Behaviour*, presented by K. Negendahl

2. Building design and building performance

"*The truth of sustainable design is that approximately 80% of the design decisions that influence a building's energy performance are made by the architect in the early design stages, the remaining 20% are made by engineers at the later stages of design.*" (Solar Heating & Cooling Programme, 2010).

Most researchers agree that decisions in the early design stage have most impact on the final design outcome. However, few building designs have been supported by early stage performance analyses, including areas of building energy consumption and indoor environment (Augenbroe, 2002; Hopfe and Hensen, 2009; Kanters et al., 2014; Wilde et al., 2001). To better qualify the reason behind the lack of integration of energy consumption and indoor environment analyses in the early stage building design, a brief overview of the motivations (and lack of motivations) is provided in this chapter. In this context the term *parametric modeling* is introduced, and on top of this the idea of *integrated dynamic models* is presented.

This is followed by a review of the state of the art in building performance evaluation and related research and practice. This includes a review of performance evaluation approaches, BPS tool couplings and couplings between BPS tools and design tools with the aim to create shared models. The chapter concludes that an integrated dynamic model, which is a model shared between a design tool, a VPL and one or several BPS tools may support the early design stage better than existing alternatives.

2.1 The motivations

The motivation to create "green and healthy" buildings is rather straightforward as seen in section 2.1.1. However, as the following section will explain the motivations are often overshadowed by the downside of financial expenses. What is interesting (or disturbing based on one's point of view) is that when building designers choose to analyze and base the design choices on consequence feedback, the incentive to create cost-efficient High-Performance Buildings is much larger. This means that financially viable High-Performance Buildings can be created. However, to achieve this, performance analyses must be part of the early design stage. Therefore, the motivation to create High-Performance Buildings is directly coupled to the motivations to include such analyses in the early design stage.

2.1.1 Threats on human health and global warming

There are many reasons why the optimization of building performance of building energy consumption and indoor environment is a good idea. Health is one concern often raised by researchers. According to the World Health Organization (WHO) 99,000 deaths in Europe and 19,000 in non-European high income countries were attributable to household (indoor) air pollution (World Health Organization, 2012). These are small numbers compared to the burden of disease due to the combined pollution of indoor and outdoor air on the global scale caused by particles from the burning of fossil fuels. According to another assessment by WHO (World Health Organization, 2006) more than 2 million premature deaths each year can be attributed to the effects of urban outdoor air pollution and indoor air pollution. More than half of this disease burden is borne by the populations of developing countries.

Buildings are responsible for around 40% of the total European energy consumption, equivalent to 36% of the fossil fuel-based CO_2 emissions in the EU (Laustsen et al., 2011). The international differences in energy consumption are very large which is why some countries may waste much more energy on average than others. Improving the energy performance of buildings is a key factor in securing the transition to a resource efficient economy and to achieving the EU Climate & Energy objectives, namely a 20% reduction in the greenhouse gas emissions by 2020 and 20% energy savings by 2020 (Laustsen et al., 2011).

Figure 6 Financial programs & incentives in Europe to optimize building performance (Laustsen et al., 2011)

2.1.2 Regulations and incentives

Design initiatives to improve energy performance and indoor environment such as increased insulation and more efficient ventilation systems are costly for the client. Therefore, to ensure that clients are motivated to invest in improved building performance, new buildings are subject to a multitude of regulations, laws and rules that aim to change buildings into High-Performance Buildings. The

EU is using many types of incentives and regulatory methods to improve building energy performance; a few examples are:

- National minimum energy requirements[5]
- Standardized methodology for calculating the cost-optimum level of buildings
- Energy Performance of Buildings Directive (Sutherland et al., 2013)
- Requirements of Energy Performance Certificate, EPC[6]
- Independent control system for EPCs[7]

Some incentives are categorized as “distant future goals”, which are requirements that are seen as good intentions but in reality are hard to follow by all European Member States (Laustsen et al., 2011). The Member States are at very different stages of implementing these less strict requirements. An example is the introduction of “nearly zero energy buildings” (NZEBs) by 2021 for all buildings and by 2019 for public buildings (Sutherland et al., 2013). “Energy Efficiency obligation” is a scheme where each Member State has to confirm that an equivalent of 1.5% of annual energy sales is saved through energy efficiency measures. For countries like Germany and Denmark such goals may seem quite loose as these countries have their own much higher goal settings. The most well-known initiative in Europe is the “20-20-20-targets” or the “2020” targets in the Energy Efficiency Directive (European Parliament, 2009), which basically forces the EU Member States to achieve the following before the year 2020:

- A 20% reduction in EU greenhouse gas emissions from 1990 levels
- A 20% raise of the EU energy consumption produced from renewable resources
- A 20% improvement in the EU's energy efficiency

In Figure 6 it is shown that such incentives may originate from different sources and be enforced by various means. Building performance requirements are but one way the EU regulates building performance.

The main legislative instrument in Europe is the 2002/2010 Energy Performance in Buildings Directive (EPBD) (Sutherland et al., 2013). The European Union supports the measures of EPBD and other initiatives by funding and other support financing programs such as European Local Energy Assistance (ELENA) (European Investment Bank, 2010). ELENA support covers a share of the cost for technical support that is necessary to prepare, implement and finance the investment program. Other types of funding like the “Joint European Support for Sustainable Investment in City Areas” (JESSICA) (EU, 2011) allows Member States to use some of their EU grant funding (Structural Funds) to make repayable investments in projects. Laustsen et al. (2011) has identified a variety of regulatory and planning obstacles; one of the main barriers is the speed at which EU Directives have been implemented by autonomous regions within a

[5] E.g. maximum energy consumption requirements

[6] IPC is a performance measurement system; Denmark had this since 1995, which makes Denmark one of the most regulated counties in the EU

[7] Sutherland et al. 2013 notes that the independency heightens the quality of implementations

Member State and the (bureaucratic) approval processes for building integrated renewable buildings. Evidently, removing many of these obstacles at EU and national level is of concern for the development of High-Performance Buildings. In terms of the current implementation of functional requirements the different Member States are subject to national regulations, which in many cases are far from imposing High-Performance Buildings.

Performance based requirements

Performance based requirements in the individual Member States provide standards and verification methods that relate more clearly to real world conditions and provide the necessary tools to designers. CIB[8] argues that this is an important advancement of innovation in building construction (Beth Tubbs, 2001). There are no uniform/harmonized/standardized European building performance requirements that can be easily compared and regulated across the EU. Even the term 'performance requirement' is interpreted in different ways (Visscher and Meijer, 2006). Although it is understood by CIB (Beth Tubbs, 2001) that performance based requirements are developed to *compare designs against performance criteria versus comparison to prescriptive solutions*, some countries understand it to constitute a description of desired levels of performance (Visscher and Meijer, 2006).

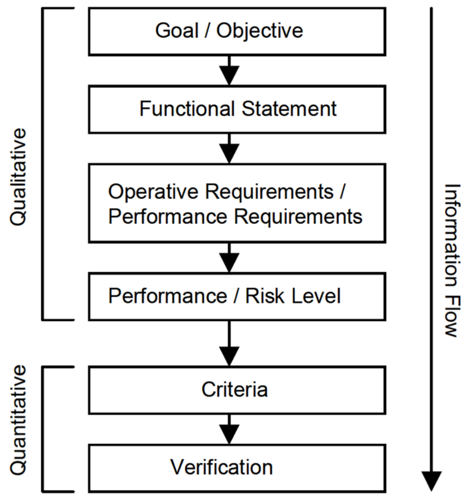


Figure 7 Performance system model based on illustration form Tubbs (2001)

Tubbs (2001) separates the requirements into qualitative and quantitative requirements. The model in Figure 7 shows that qualitative requirements are placed on a higher level in the information chain than the quantitative requirements. The main motivation for countries such as the UK for performance regulations is to keep the regulations brief and qualitative so that it would reduce

---

[8] CIB stands for "Conseil International du Bâtiment", in English "International Council for Research and Innovation in Building and Construction"

the *burden of sending the 350 page prescriptive building code through parliament* (Beth Tubbs, 2001). Also, the argument can be found in computing; prescriptive building codes may be defined by a qualitative measure, hence it needs human evaluation. If the requirement is a number, it is computable (Turing, 1936). Thus, it is far more feasible to compare computable requirements on a large scale than to compare qualitative requirements on large scale.
Performance based requirements are associated with the general threats on global warming and global health, however sometimes the arguments have a more economic dimension.
It is a well-known argument that indoor environmental performance such as air quality and thermal environment is correlated with "human performance" (Edwards and Torcellini, 2002; Seppanen and Fisk, 2006).

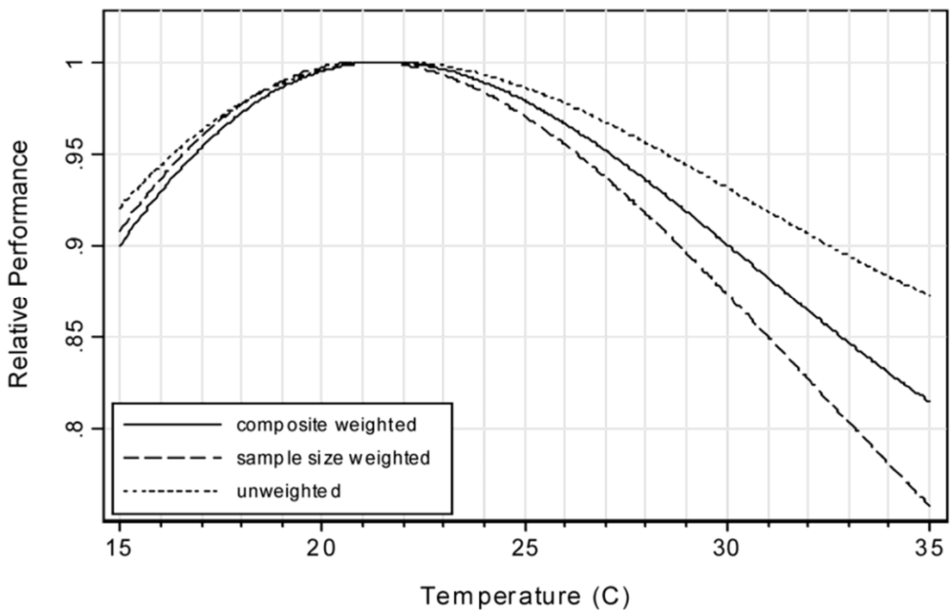


Figure 8 Relative "human performance" vs. indoor temperature, from (Seppanen and Fisk, 2006)

The human ability to concentrate, thus being productive is deeply associated with how we feel, and greatly affected by environmental changes such as temperature (see Figure 8. Regulations have sought to optimize working conditions by e.g. imposing stronger requirements for office buildings than for residential buildings, especially in areas of thermal requirements and indoor air quality; this is seen, for example, in the Danish building regulations (Danish Building Regulations, 2013). The reasoning behind these regulations is therefore based on national economy and national health rather than on global health and environmental concerns.

In Denmark, performance based requirements for energy consumption have been regulated by national regulations for decades; and, few other countries have pushed regulations on building energy consumption as far as Denmark over the past 10 years. As mentioned, future regulations are pushing the requirements towards High-Performance Buildings such as the initiative of NZEB, and aim to protect the indoor environment with various other performance based requirements. The tendency is that performance based requirements are used in regulations where it is possible to predict the performance outcome of the building (or system / component) in relation to its context, its use and specific design (Struck, 2012). Prescriptive specifications are used where evaluation is difficult, for example in cases like “social needs”. They may also regulate cases concerning human safety based on historic fatal situations. In comparison to performance based requirements, prescriptive requirements are argued to have the inherent potential to hinder change and innovation as they prescribe solutions (Struck, 2012). Even after the transition from prescriptive requirements to performance based requirements in national regulations, the freedom for innovative design concepts is still restricted by local building regulations, the design brief and other client prescriptions. The design brief can in its most strict form prescribe design solutions, whilst local authorities demand compliance with building regulations such as special attention to shore line and animal protection. In addition, Laustsen et al. (2011) mentions that regulations are rarely enough to promote further improvements of building performance than the minimum requirements of the regulations. Therefore, private initiatives and project specific focus on building performance are important to reduce carbon emissions and improve public health. It all comes down to the motivation of optimization of building performance.

2.1.3 Motivations for High-Performance Buildings

Whether or not the motivations for developing High-Performance Buildings arise directly from the global threats on human health and the environment, or by economic incentives such as improving indoor climate to make people more efficient, or reducing the dependency of oil, gas and coal from other nations, are unclear. Some will argue it does not matter, as long as the objective to succeed in building High-Performance Buildings is fulfilled. There are many examples of High-Performance Buildings, buildings that have been designed with the aid of simulation tools and subject to dedicated design principles and efficient methods. However, none of these design principles, methods and tools has efficient means to consider and include the many other aspects of design (see Figure 9) beside those that promote High-Performance. *Why is that? And why is it at all interesting?*

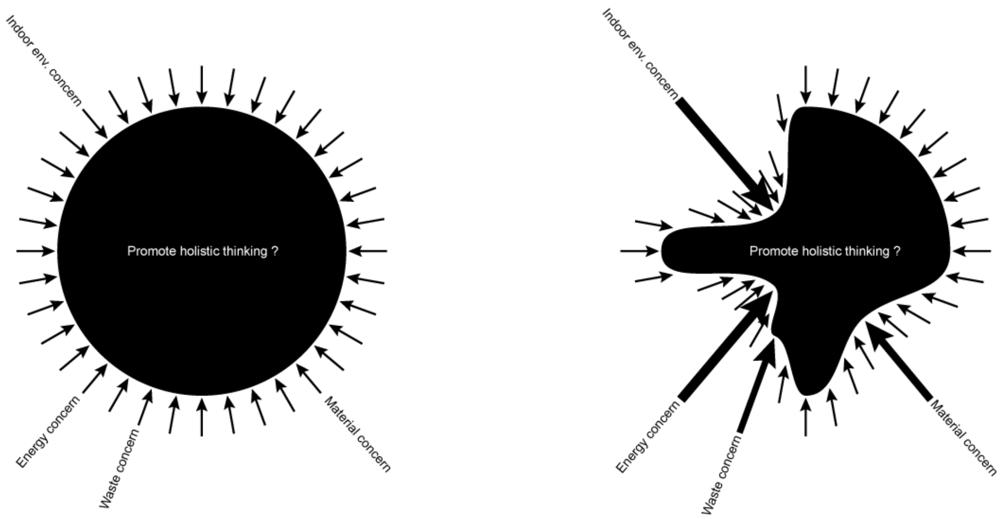


Figure 9 How do we maintain holistic thinking in design in practice, when outside pressure on the design stage is unevenly balanced?

Many architectural studios and consulting engineers today claim to be on the frontiers of sustainable design. Some have even posted clearly defined design paradigms or ethical etiquettes on their websites for the world to see.
The mismatch between the good intentions / motivations and the buildings that have been built in the last decade is quite staggering. Therefore, it is interesting to discuss why building regulations are far from enough (Laustsen et al., 2011) to ensure High-Performance Buildings and why 30% of the modern building mass does not provide a healthy indoor environment (EPA, 1991). One explanation of this is cost. If the client prefer to lower construction (capital/investment) costs on the expense of running (operation, maintenance and repair) costs, all the blame can be reverted to the client. However, the real explanation is more complicated than this.

The cost of change

It is intriguing and thought provoking that building evaluations of life-cycle costs, building energy consumption, and indoor environment etc. have been omitted from the early design stage in almost every building design project to this day. Analyses are immaterial. They only depend on time and human resources, and unlike "a more efficient ventilation system or increased insulation" analyses are costly and do not produce a physical object for the client. The client must be convinced that better building design has a physical value. If the client believes that these analyses will improve the building, he must invest carefully. The best investment in the eyes of the client is the "cheapest way" to change the building to be more cost efficient, green and healthy. Somehow, this "cheapest way" is declined, forgotten or for some reason not grasped when it is at hand.

One way of thinking of the missed opportunity is to measure design changes in terms of cost. This is exactly how MacLeamy (2013) approaches the value of buildings. Changes are expensive, and reducing the amount of changes will reduce the cost of designing a building. But changes are not equally expensive during the development of the building; the relative cost of a change is smaller in the early design stage. Therefore, MacLeamy's principal argument is that one must *"seek to contain the changes as early in the design stage as possible"* (MacLeamy, 2013).

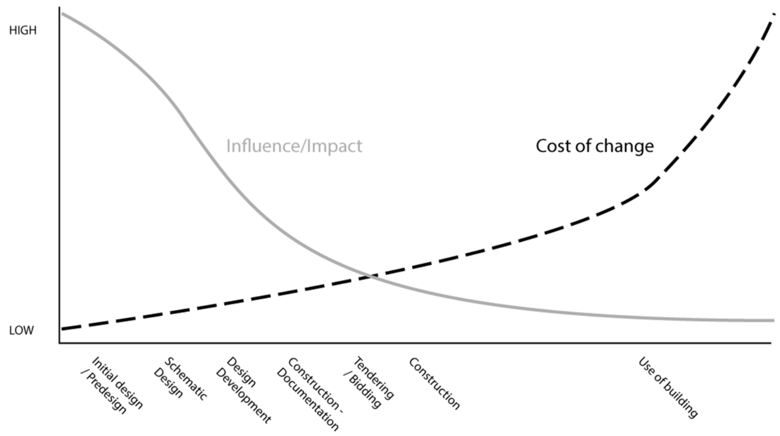


Figure 10 The relationship between influence in building design and cost of changes. Based on Boehm (Boehm, 1984), Beck (Beck, 1999) and Paulson (Paulson Jr., 1976)

Davis (2013a) argues that most building designers would perceive their decisions (and reviewing decisions, which is a change) as adding value to a project. Therefore the *change = cost paradigm* is rather crude. Davis further notes that the shift in effort then assumes that the design process can be anticipated and the design problem can be known before the design commences, which of course is very unlikely. Change is for these reasons not necessarily a cost to be avoided. What is important in this discussion is acknowledging the potential value in the choices of early design stage decision making. MacLeamy was not the first to see this. Davis (2013a) tracked the "hidden potential" through software engineering (Boehm (1984) and Beck (1999)) back to the building industry with Paulson (1976). Paulson (1976) talks about the benefits of making early decisions when the designer's level of influence is at its highest level. And even in 1976 Paulson describes the idea as not being new and points backwards toward manufacturing and heavy-industrial design (Paulson 1976). The question is: To this day, after four decades, why have so few practitioners focused so little on the early design stage potential?

Back in 1976 Paulson points towards changing the contractual arrangements. Subsequently "knowledge injections" in the early design stage would be valued higher, which would increase the level of influence and thus the potential of change.

"A main barrier for cooperation in the early design stage is economy. Handling engineering performance aspects (in the early design stage) increases the workload which is not compensated in fees." (Petersen et al., 2014)

In a recent survey among Danish architects Petersen et al. identified that fees do not to this day reflect "knowledge injections" in the early design stage. MacLeamy (2013), who is part of the Strategic Advisory Council for the BuildingSmart (BuildingSmart.org, 2013) initiative, advocates both the format Industry Foundation Classes (IFC (BuildingSMART, 2013)), and process Integrated Project Delivery (IDP). Here, IFC is considered one of the two most commonly used open formats for building model exchange (the other is gbXML (gbXML.org, 2014)). IDP is a process that contractually binds project parties (including the members of the design team) towards viable solutions early in the project. What is interesting in MacLeamy's advocacy is the dedication towards IFC, and the clue which lies herein. Contractual and legal commitments are not the entire answer, but model exchange is a key to unlocking early stage building performance potential. Section 2.4.5 deals with the model exchange and discusses IFC and other means of improving model coherency.

What further complicates the unlocking of the hidden potential is the circumstance that decisions have to be made with limited resources on the basis of limited knowledge. This is partly due to the nature of information levels (Hermund, 2009a) where information is created and not available before it is made into a definite decision and partly due to the quality assurances not being upheld (Hensen, 2002). The latter concerns quality assurances as mentioned by Hensen (2002), although these will evidently be solved if those who design are experts in BPS tools or if the design team includes simulationists (de Souza, 2009).

2.1.4 Motivation for parametrical modeling

Parametric modeling has opened up for new possibilities for building design, enabling novel geometries to be generated in a relatively short time span (Harding et al., 2012). These models are often created in a graph representation, in a Visual Programming Language (VPL). This enables the user to construct associations between parameters and geometrical functions in order to quickly model designs within a set of constraints. Adjusting parameters is a top-down approach, where the intention of the model is represented in the model itself. And the process of adjusting parameters relies on the feedback from the model.
According to Davis (2013a) the introduction of parametric modeling was motivated by a desire to decrease the cost of change:

"In theory, a parametric model helps lower the cost of change if the manipulation of the model's parameters and explicit functions rebuilds the geometry with less effort than would otherwise be required from a designer." (Davis 2013a)

Parametric modeling would arguably not only reduce the cost of changes but change the whole concept of how to design. Currently, parametric modeling is being used to analyze everything from building envelope, form, and core, to material design and comparisons to structures found in nature (Courtney L. Fromberg et al., 2015). Marcello and Eastman (2011) argue that expertise within the field of architecture is strongly based on "rules of thumb" from trial and error from previous experience. And parametric objects can be developed to *nest what has already been learned* in order to create the parameters to generate more concise solutions. Davis (2013a) reasons that front-loading changes such as argued by Paulson (1976) and MacLeamy (2013) may not the best way to deal with design: Rather than making decisions early in order to avoid the expense of changing them later, the cost of change can be lowered (when parametric modeling is applied) *to the point where critical decisions are delayed until they are best understood* (Davis 2013a).

Parametric models and flexibility

The key in parametrical modeling lies in the flexibility of the model. By maintaining a highly flexible model the designer can afford to make more changes. While some changes can be anticipated and as Davis (2013a) notes; *perhaps even front-loaded, many changes come from forces outside the designer's sphere of influence*. This is especially true if the designer represents a design team with multiple actors with multiple objectives. These actors may seek changes in opposite directions, which will lead to many potential changes. Coming back to the information levels other changes occur because design is a knowledge evolving process. This means that if a model is sufficiently flexible it may accommodate changes spanning multiple levels of information. Given a high level of flexibility the parametric model can absorb new information and ongoing decisions.

> *"If we could make the parametric model flexible enough …, the design process becomes almost inherently more "designer-ly". The designer could delay design decisions until they can best decide what the impact of those design decisions are."* (Davis et al., 2014)

In theory, this means that a model may be built with high flexibility on several levels of scale. The large scale lets designers change location, orientation and large geometrical features such as number of stories. The smaller scales may be number of windows, window properties or detailing on façade systems. The ability to absorb and distribute information throughout the design process is game changing, since early decisions are just considered as any other variable. As a consequence, the building designer does not need to make a complete remodeling of the design. More importantly, the option to actually change early decisions based on knowledge acquired at a later stage is key to improve the building performance in the early design stage.

When talking about parametric modeling, it may be necessary to distinguish between the implications of the modeling approach and the way in which parametric modeling is approached. There are many ways to approach parametric modeling in architecture and engineering, but they all involve the use of an algorithm. An algorithm is often regarded as consisting of a logic component and a control component. The logic component specifies the knowledge to be used in solving problems, and a control component determines the problem-solving strategies by means of which that knowledge is used (Kowalski, 1979). Furthermore, the logic component determines the meaning of the algorithm whereas the control component only affects its efficiency.

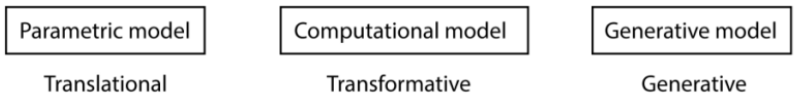


Figure 11 Variations of parametric models

Parametric models may be categorized in different ways. Tamke et al. (2013) categorizes the type of models according to their complexity (seen in Figure 11) or how the model can serve an *emergent* design process. The most direct way of approaching parametric modeling is by using basic parametric models that are defined by parameter translations via an algorithm. Computational models, which are given explicitly formulated incremental intelligence, may change parameters by automation and feedback processes. The more advanced uses of parametrical modeling are associated with generative modeling techniques. This last variant includes some kind of recursive intelligence in the model, which means that the model is allowed to change its own way of functioning. Parametric modeling with the inclusion of optimization techniques is categorized as computational models or generative models. This is further discussed in Chapter 3.

Parametric models and the use of VPLs do, however, not come without costs. VPLs such as Grasshopper (Robert McNeel & Associates, 2013b) and Dynamo (Autodesk, 2013a) are defined as Graph-based programming tools. The most popular way (today) of constructing parametric models is through Graph-based models by the *one-graph, one-model approach* (Harding et al., 2012). Here, the user must choose how to set up a parametric model early in the process. However, design requirements and performance evaluation criteria are likely to be subject to change as the design process develops. Harding et al. (2012) argues that in practice this can become a problem; as the complexity of the parametric model increases, the dependencies become more difficult to adjust, and design freedom actually decreases. As a result, the initial parametric relationships tend to get ‘locked-in’ and cannot adapt. This is especially true when graph-based parametric models are used in combination with optimization algorithms. This again is discussed in Chapter 3.

2.1.5 Parametric modeling and simulation feedbacks

With the introduction of parametric modeling the influence of building performance has moved into the early design stage, and given the high flexibility of the models it also means that more decisions can be made in later stages. The shift is only possible because of the parametric properties of the model and the way decisions are considered less definite compared to traditional modeling. The cost to make changes is practically reduced to nothing and the cost of changes remains low further into the design process. Now the argument may be taken a step further. As integrated dynamic models are parametric by nature they facilitate the same properties concerning the influence on the design process. If integrated dynamic models are built to accommodate high flexibility, the flexibility is transferred to all the tools dynamically coupled to the model. An integrated dynamic model will therefore not only provide consequence feedbacks from one or more dynamically coupled BPS tools; it will provide this with the same parametric properties as any other parametric model. This creates the opportunity to review early decisions in later stages in the process based on performance feedback. Thus, it opens up for much more goal driven processes as the BPS tools facilitate clear performance metrics to evaluate and compare.

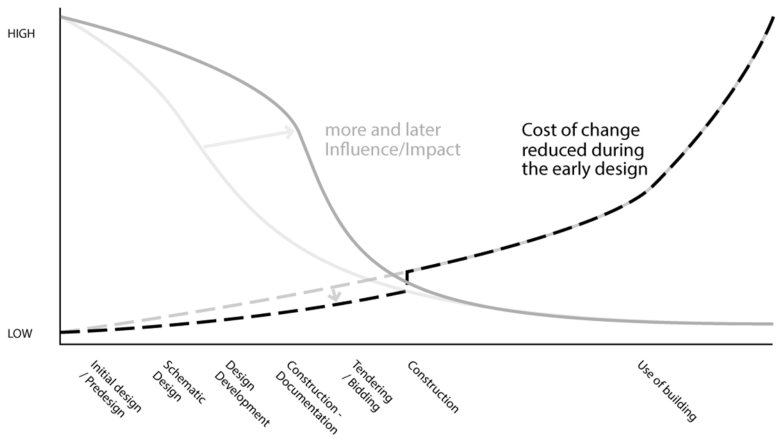


Figure 12 The relationship between influence in building design and cost of changes by integrated dynamic models. It is hypothesized that the integrated dynamic models will shift the building designers' impact level into and thus reduce the cost of change in later stages of the design process

Given the high model flexibility, the design team may open up much more design variables that may or may not directly influence the performance, but it will definitely influence the design process. Without this option to freely open design variables the design team needs to lock the design in early on just to have time to analyze the performance of the design. This is particularly true if the design team uses one tool to design and another to simulate. The integrated dynamic models enable more analyses for a longer period of time, which in essence makes each performance analysis less costly.

It all looks very promising. The design team needs to utilize parametric modeling with integrated dynamic models; this increases the potential capitalizing on the reduced number of the more expensive changes. This, however, is not the entire story. Parametric modeling does not come without costs itself. The figure seen above (Figure 12) shows the cost of *a* change and does not include the cost of investment in parametric modeling.

As mentioned on several occasions, the design has to be locked in an earlier stage if parametric modeling is not used. This again means that there is room for fewer changes in the traditional (non-parametric) process. To compare the parametric process with the non-parametric process we need to estimate the total cost by multiplying the cost of individual changes with the number of changes. If the total cost of changes (in the early design stage) is lower for the parametric process compared to the non-parametric process we have a business case. In such case this would include a “financial viable motivation” for the design team to dedicate more time and effort to improve building performance.

2.1.6 The motivation for Consequence based design

The cost of shifting from manual, traditional, non-parametric processes to a complete streamlined parametric process is difficult to estimate. But let us think of it this way: If the design team chooses to lock their design early on and agrees to make as few changes as possible, why should the design team invest in an integrated dynamic model that allows them to change more and in later stages? Economy is still the key issue, but now it is the design team’s economy that is at stake, not the life-cycle costs of the building. To take on this challenge, the design team has to be economically motivated to carry out changes at an early stage, as concluded by Paulson (1976). Even though some may argue that motivation is not always driven by economy and the actual potential to make High-Performance Buildings. Then followed by the argument: that the economic motive is redeemed if the client is convinced to pay more for High-Performance Buildings. Then it is quite clear that given the lack of economic motivation Consequence based design[9] is difficult to put into practice.

If we are to pursue High-Performance Buildings the containment of changes in the early stages might therefore not be enough. Minimizing the amount of changes also means minimizing the amount of variations, alternatives and iterations. The potential of finding higher performing buildings is correlated with the amount of recursive iterations for example as found in an optimization process. Therefore, it must be assumed that more evaluated alternatives available in the early design stage means higher potential to “find” a High-performance solution. For this reason, if the design team (indirectly the client) values the idea of creating High-Performance Buildings, they need to accept more iterations and the relatively larger effort to evaluate more solutions.
Therefore, it is natural to argue for a “top-down”, this is exactly what Bernal and Eastman (2011) does:

> *“Design activities vary from high-degree of freedom in early design stages to highly constrained solution spaces in late ones, which entail large amount of design*

[9] The Consequence based design approach is defined by the explicit creation and utilization of integrated dynamic models in the early design stage (as described in 1.4).

expertise. A top-down approach based on nested assemblies and custom functions is proposed to embed such a design expertise in reusable parametric objects." (Bernal and Eastman, 2011)

This approach is really valuable if all members of the design team follow the same pace when moving from one design stage into another. And this requires much coordination and most likely also an economic incentive, as Paulson (1976) argues. This could be based on contractual agreements punishing those who fail to confine changes in early design stages, or it could be based on a shared (by the design team) economic motive to generate larger profits. The answer to such a motive is found in the economy behind the parametric modeling or the efficiency of automating analyses; however this is complicated and difficult to communicate to a client. Return of investment (ROI) is, however, easier to communicate.

The accumulated price of the construction and management of the building can be described in the same way as seen in the figures above (Figure 10 and Figure 12), but this time we look at the entire life cycle of the building.

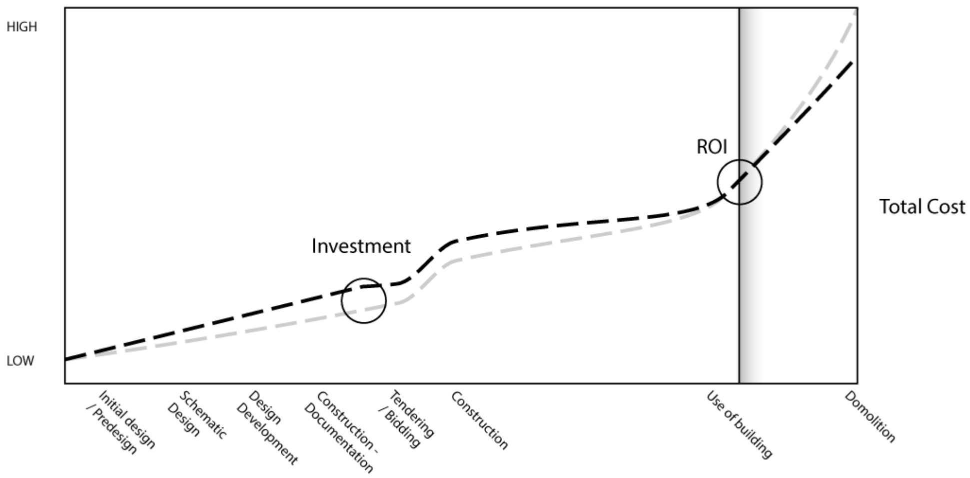


Figure 13 Return of investment (ROI). In traditional modeling the investment lies in improved, but more expensive solutions (primarily on system level)

In Figure 13 ROI is placed somewhere in the life cycle after the building is taken into use. How good the project is in terms of investment may be defined as the point where the return of investment is placed in the building life cycle. The investment may be energy savings, cheaper materials, and more functional solutions that will make the process of maintenance less expensive. It may also be based on more qualitative performances such as increased social value. These are simply improvements of investments (or changes) of the yet to be built building, and therefore possible to implement in the early design stage. ROI implies an investment that many clients would consider as an additional expense, which is why late ROIs are harder to implement in practice. Before integrated dynamic models and parametrical design can really pay off, the concept of Consequence based design must consolidate an early ROI.

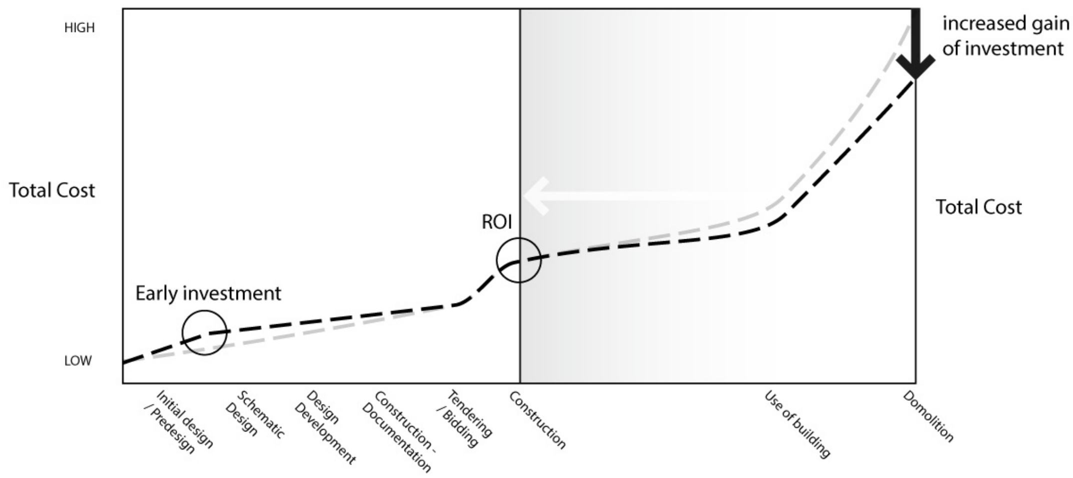


Figure 14 The introduction of integrated dynamic models. The investment is primarily focused on better systems and more thoughtful solutions, but not necessarily more expensive solutions. The integrated dynamic models ensure more analyses and earlier ROI.

In Figure 14 it can be seen how the integrated dynamic models can be regarded as investments. Here the argument is that more analyses (we are talking about a magnitude of 10 to 1000 more analyses opposed to traditional modeling processes) in the early design stage, will shift the ROI into a more favorable position.

In theory, the ROI may be shifted all the way into the construction stage. However, this is only possible given the models has a high flexibility and given design team has the right quality assurances (Hensen, 2002). Given these circumstances the client's investments in integrated dynamic models will be "free" and High-Performance Buildings may be reduced to the cost of a traditional building.

Now the motivation for application of integrated dynamic models and the approach of Consequence based design has been established. This thesis, from this point on, will discuss how integrated dynamic models are to be applied in theory and in practice.

2.2 High-Performance Building analysis in the early design stage

Following the previous section on the motivations of the design team to design High-Performance Buildings, it is necessary to review how other researchers have approached this objective and the reasons why High-Performance Buildings are so difficult to design. Past research may be separated into three approaches of how to integrate building performance considerations in the early design stage. These three approaches are driven by:

· Regulations
· Methods
· Tools

Regulations cover the performance-based requirements which are to various degrees law enforced as described in 2.1.2. Methods cover design frameworks consisting of methods and guidelines such as the integrated design process to commercial certification systems (see 2.3), and Tools cover design and performance tools see 2.4. Most focus will be given in this review and in the thesis as a whole to the "tool approach", while the "method approach" will also be discussed. In many cases, however, these two approaches overlap as methods may prescript tools and some tools may only be used in certain methodological ways. The "regulations approach" will not be discussed further.

A shared argument behind these methods and tools are that they will be beneficial if used as intended, thus creating benefits as illustrated in Table 1.

Table 1 Benefits to owners and occupants of High-Performance Buildings from (Bienert et al., 2012)

Clients (owners)	Users (occupants)
Reduced operating costs	Reduced operating costs
Enhanced brand	Enhanced brand
Mitigation of future regulatory impacts	Mitigation of future regulatory impacts
Reporting to stakeholders	Reporting to stakeholders
Tenant retention	Employer of choice, employee retention
Increased rents	Enhanced building (indoor) environment
Differentiated position of asset	Improved productivity
Increased market share	Decrease their footprint on the planet
Higher net revenue return	Stronger tenant/owner/manager relationship

2.3 Design methods

While European and national performance based requirements are still far from implemented in many countries (Laustsen et al., 2011), private players (investors) have taken initiative to stimulate clients to embrace energy and material conservation, improve indoor environments and design towards more efficient sustainable futures. Here, rating-schemes are used widely and are available with different scope and for different phases of the building's life cycle (Hopfe et al., 2005). Rating-schemes such as LEED (http://www.usgbc.org/ leed), BREEAM (http://www.breeam.org/) and DGNB (http://www.dgnb.de/) (and variations of this system such as the Danish DGNB-system (http://www.dk-gbc.dk/)) are voluntary and privately founded rating-schemes awarding labels certifying a degree of building performance following the design and construction phase. In addition to the certification systems mentioned above this review only covers the integrated design method. However, there are similar design methods that attempt to encompass goals of High-Performance Buildings (for instance the Swedish system Miljöbyggnad (https://www.sgbc.se/)), and while they might deviate from the methods described here, they are presently not used in such a scale (in Denmark at least) that they are worth mentioning.

2.3.1 Certification systems

The aim of the rating in certification systems is to provide transparency and subsequently increase the monetary value of the rated building by showcasing its performance (Bienert et al., 2012). For the early design stage, LEED, DGNB and BREEAM claim to assess the overall environmental impact. These types of certification systems all share a common way of handling non-legislated and unquantifiable factors: they are defined by rules. In most cases, elaborate "book sized" rules are needed to cover the system. The rules call for a certain insight into the system to make evaluations consistent from one project to another; this justifies the requirements for capable and educated specialists handling the system. The certification systems all have a price mark attached and are developed to *certify* a building project. Most certification systems provide a pre-assessment methodology that certifies the buildings in the early design stage. These assessments are often lightweight versions of the later assessment procedures, but in general function in the same way. To make a pre-assessment, educated assessors are required and the methods have to be followed as described.

2.3.2 Integrated design process

Instead of utilizing a fixed rating system more loosely defined methods have been developed with the aim to achieve similar goals. The Integrated design process (IDP) widely known in Denmark and internationally was developed in the period of 1997-2002 by the IEA Solar Heating and Cooling Programme, Task 23 (Löhnert et al., 2003). The outcome was a design process targeting the early design phase and the development of methods and tools to be used by the design team. The design team is introduced as a collaborating team of architects and engineers as well as owners, contractors and building users. The process initiates the use of design loops where the design team establishes a number of different alternative solutions. The solutions are each investigated in detail and only the best performing solution is taken through to the next iteration or design loop. This performance-based framework has proven to be highly successful in producing high-performance and environmentally friendly buildings.

The criticism of the method is that it fails to describe a rational approach to set up the initial alternative solutions and that the approach based on numerous iterations can be inefficient and time consuming. These shortcomings were later improved in various studies. Latest is the effort seen with Petersen 2011 which seeks to introduce integrated design as a performance-based design paradigm thinking in the early design process. This elevated the definition and choice of predefined performance based criteria into an important part of the design process. The performance-focused paradigm helped the design team to find and improve solutions that met the design criteria and minimized investigations of solutions that would not fulfill the performance criteria. This approach reduced the time and effort needed to design high-performing sustainable buildings. The paradigm change illustrated the circular/iterative thinking in the Integrated design process (design-simulate-evaluate-design…) and made it accessible for design teams to investigate numerous design concepts over a short span of time. The barriers of the approach were found in the implementation of design- and simulation environments (i.e. iDbuild, Petersen & Hviid 2012) and the *process-related issues* of the approach, such as the need of the simulationist[10] to explain the results to the building designer (interdisciplinary acceptance of roles) and the building designers ability to accept and make use of simulation-based design support as a design driver (Petersen, 2011). While the dedicated simulation tools such as iDbuild was found efficient to evaluate building performance in the early design stage, the inclusion of qualitative objectives such as architectural quality was only vaguely considered in the method, in the tool and the model. Which meant the design problem definition is clearly created by the *indisputable* boundaries of performance based criteria and evaluations. Within these boundaries other design objectives, including architectural quality has to be found. As the tools accommodate this rationality the idea of building performance criteria defining the boundary conditions of the final design is easy to understand. However most building designers perceive building design as a solution to a magnitude of objectives, many of which are qualitative and hard to assign boundaries to, and many of which will change during the design. Using a performance-based solution space as a continuous/dynamic design driver or as a fixed boundary condition is a very different approach. The essence is qualitative objectives are difficult to include as a rational design approach and implementing such objectives in tools is even more challenging. As a consequence architectural quality had been disregarded since the beginning of the formulation of Integrated design in 1997. Löhnert describes the issue regarding architectural quality as a "problem" almost impossible to "solve":

> *"Architectural quality however, leads to an almost insoluble problem, since this requirement is exclusively based on project-specific evaluations and is therefore very strongly dependent on the intuitive, cognitive and aesthetic factors put forth by the individual participants."* (Löhnert et al., 2003)

The integrated design method's solution is simply to find the boundaries of performance; performance-based design processes as suggested by Petersen

---

[10] It is acknowledged by many researchers that projects are increasing in complexity and cannot be realized without involving experts at the very early stage of the design process. Augenbroe (2001) argues that no architectural studio would take the risk of relying on designer friendly analysis tools, because it will take a high degree of expertise to judiciously apply simplified analyses to complex buildings. Therefore, to completely integrate tools one must integrate the expertise of people.

2011 is one of the resent progressions of this rationale. How, and to what degree qualitative objectives are fulfilled remains within the Integrated design method undefined.

2.3.3 Performance and design support

"As we are progressively moving towards dispersed teams of architectural designers and analysis experts, full integration of all disciplinary tools in a collaborative design framework is the ultimate goal." (Augenbroe, 2001)

Integrated design as a process, as certification standard or as a method is widely used today. These more or less prescribed methods require levels of *collaboration* or *integration* between the building designer and the simulationist. The integrations can be divided into three domains which are discussed thoroughly by (Negendahl, 2015a):

1. Human domain
2. Tool domain
3. Model domain

The existing methods (the Integrated design process and certification systems) require large amounts of interaction in the human domain to ensure performance analyses are used correctly and effectively in the building design process. Augenbroe cited above talks of the "ultimate goal", which is not about integration of people but the integration of tools. Even though non-collaborating building designers and simulationists are unlikely to achieve any form of High-Performance Buildings meaningful and beautiful for real people Hermund (2009a) the idea of integrating the tools is extremely important. To follow Augenbroe's lead the discussion of integrations will revolve around the integration of tools (Section 2.4) and the integration of models (Section 2.4.5).

Until now, performance based requirements have been regarded as fixed, predefined and governed by regulatory authorities, based on certification systems or clients' own objectives. Additionally, the performance based requirements such as Löhnert et al. (2003) related to the goals of creating High-Performance Buildings and Integrated Design; they are defined by the reduction of resources and improvement of the indoor environment. Nevertheless, performance may be evaluated by other means in addition to these two categories. Building performance evaluations can be conducted at different levels of abstraction and for a range of other requirements. One example is (Borden, 2008) who applies performance evaluations in the human centered research field by evaluating interaction of humans with their natural, social and built environment. Mallory-Hill (2004) maps and evaluates building performance (of work spaces) based on different demands such as strategic, ecological, functional and economic. These demands are further broken down into the dimensions of building system levels (e.g. space, skin, structure and site) and architectural system levels (e.g.

workspace, floor area, building and built environment). The argument is that these requirements reflect client expectations of the final building performance.

What unites the different views on performance based evaluation is the need and the ability to quantify the various performances in a systematic way. To quantify building performance during an evolving design process, stated requirements need to be converted into performance indicators and metrics (Struck et al., 2009). A performance indicator is thereby defined as an *objectively quantifiable performance measure* describing the building performance in order to support dialogues between stakeholders in the design process or to document given requirements towards authorities. A performance metric is a quantity that has three distinct characteristics according to Deru and Torcellini (2005):

- Measurable
- Clear definition of end goal
- Clear definition of boundaries

Following this definition, building performance based metrics can be any building related variable and is only comparable (and computable), when all three characteristics are present.

2.4 Design and performance tools

There are no clear definitions of a tool that can deliver performance based feedback in the early design stage. To better cover the design and the performance aspects of tools that fall within the category of "design *and* performance", this section first introduces the most frequently used categorization of tools: "designer-friendly tools and design-integrated tools". Then the most important requirements of the BPS tools (solvers), and integration approaches behind these tools are reviewed. The solvers are mentioned to make the reader aware of some of the underlying technical challenges in simulation and why some BPS tools are hard to implement and utilize in the early design stages. This is relevant in relation to some of the developments of new BPS tools discussed in 4.4. Finally the BPS tools are discussed in relation to the integrated dynamic models.

The interest of "performance" in this thesis is related to the High-Performance or "green and healthy" buildings as defined in the introduction. Performance evaluation, however is often composed of several sub evaluations as seen in Figure 15. Therefore, in typical design analyses more than one BPS tool is often needed to evaluate the overall performance of the building design. This is why BPS tools are mentioned in plural.

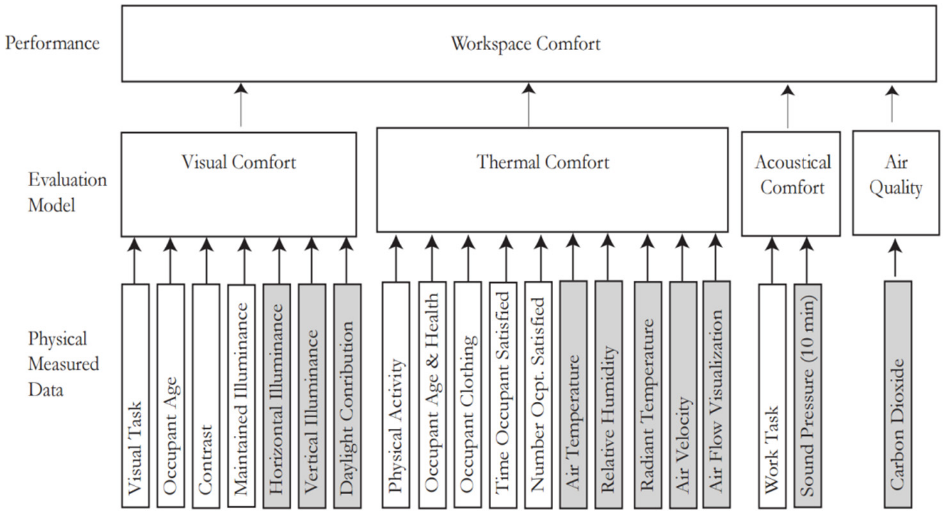


Figure 15 Evaluation of "comfort in workspaces" based on several sub evaluations, such as thermal comfort, visual comfort, acoustical comfort and air quality, from (Mallory-Hill, 2004). Grey physical measured data markers are more often used than white data markers.

2.4.1 Designer-friendly tools versus Design-integrated tools

The basic distinction between designer-friendly tools (Figure 16, left) and design-integrated tools (Figure 16, right) is the reduction and encapsulation of domain knowledge in the first case versus enrichment and externalization of design context in the second (Augenbroe, 2002).

According to Augenbroe (2002) the once popular research area of designer-friendly tools depicted in Figure 16 to the left, seems to have been replaced by the strategy of integrating design tools, illustrated to the right. By contrast to its name, the integration is meant to delegate ('outsource') design analysis to domain experts and their increasingly complex expert tools. The latter concentrates on an efficient communication layer that supports the delegation of tasks and interpretation of results. Whereas designer-friendly tools emphasize the import of 'packaged' domain expertise into the design team, design-integrated tools emphasize the *export of formalized analysis requests along with an explicit design context* (Augenbroe, 2002).

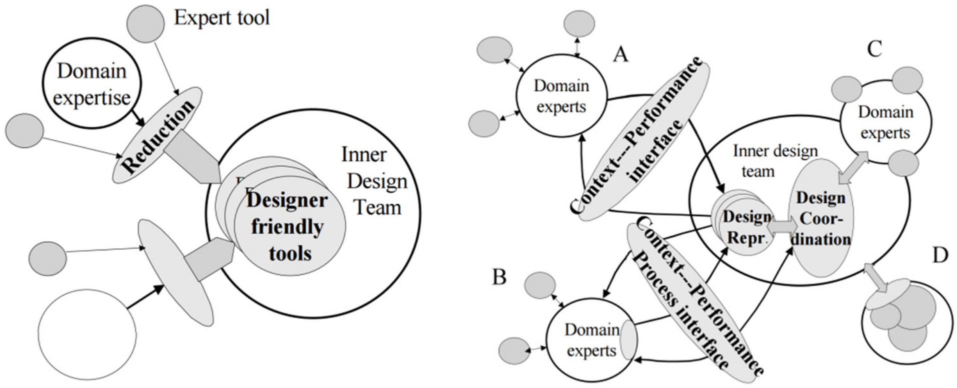


Figure 16 To the left the encapsulation of domain knowledge is placed in "Designer friendly tools". To the right the encapsulation remains separated but correlated in "Design-integrated tools", from (Augenbroe, 2002)

Equally important is the backpropagation of analysis results in a form that supports better informed rational decision-making. This has repercussions for the way in which the design team operates. Instead of a tool user, the inner design team needs to become a central design manager, maintaining a central design repository and acting as a coordinating agent for domain experts. While Augenbroe (2002) mentions variations of the design-integrated tools, they rest on the assumption that these environments will ultimately be sufficiently transparent to be accessible to members of the design team without a significant reduction of domain expertise. This assumption takes us back to the origin of designer-friendly tools. Later in this chapter, when the definition and origins of integrated dynamic models are established the reader might notice that both the definitions of designer-friendly tools and design-integrated tools may be applied to integrated dynamic models.

When considering computational support for performance evaluation computational methods[11] must distinguish between the design and the operational phase of buildings. While building performance can be evaluated by measured data to the design specification, this is obviously not possible during the design stages. ASHRAE (Ahsrae, 2005) differentiates between these two main methods as a:

· Forward approach
· Data-driven approach

2.4.2 The Forward approach – The BPS tools

This approach presumes detailed knowledge of the natural phenomena affecting system behavior and the magnitude of interactions (effective thermal mass, heat and mass transfer coefficients, solar transmittance etc.). These methods are most often implemented as solvers in large scale software tools categorized as building simulation (BPS) tools. A BPS tool is defined by its ability to predict the output

[11] Computational methods are sometimes called "solvers" when implemented in BPS tools

variables of a specified model with known structure and known parameters when subject to specified input variables (Ahsrae, 2005).
There are two basic types of BPS tool methods that dominate the Forward approach (ISO, 2008):

1. Quasi-steady-state method
2. Dynamic method

The quasi-steady-state method is based on heat balance calculations over a sufficiently long period of time (typically one month or a whole season). This enables designers to take dynamic effects into account by an empirically determined gain and/or loss utilization factor.
The dynamic method is based on heat balance calculations within short time steps (typically one hour) taking into account the heat stored in, and released from, the mass of the building.
The two basic types of simulation methods are different in many ways. They divert in terms of accuracy, flexibility and speed in the delivery of results. In general quasi-steady-state methods are much faster and less complex, but also less accurate than the dynamic methods. Clarke (2001) suggests that as a general strategy, it seems reasonable to aim for a high level of accuracy combined with a model structure that is capable of adapting to the information available at any design stage. However, as he remarks:

"It is impossible to establish, a priori, the optimum level of model accuracy and flexibility in the field of building energy simulation. Indeed, the trade-off between accuracy and flexibility is itself a dynamic concept that will vary according to the modeling task in hand." (Clarke, 2001)

Therefore, there is no best-practice definition of the choice of BPS tool (solvers) to lean against, when attempting to use simulations in the early design stage. However, there are a growing multitude of choices between tools, and various researchers have tried to better categorize the requirements of BPS tools for the early design stage.

Design requirements of BPS tools

A great number of tools claim to support the evaluation of building performance. The Building Energy Software Tools Directory (U.S. Department of Energy, 2015) lists 419 tools for the evaluation of energy efficiency, renewable energy and sustainability in buildings. Between 1997 and 2010 the number of tools has almost quadrupled (Attia et al., 2012), which tells us that there are plenty and a growing number of options available for designers and design teams to utilize BPS tools in the various design stages. According to Augenbroe (2001) the list reveals that the emphasis has shifted from an early focus on energy consumption to many other building performance characteristics. The hundreds of man-years that have been invested in building performance analysis tools have paid dividend. It is now possible to choose from a range of tools in each pertinent performance domain.

A number of studies and surveys have been carried out in the past in the field of criteria and requirements of BPS tools. Attia et al. (2012) has identified five top-ranked requirements with the aim to find a common benchmark for BPS tools:

1. Usability and Information Management of interface
2. Integration of Intelligent design Knowledge-Base
3. Accuracy of tools and Ability to simulate Detailed and Complex building Components
4. Interoperability of Building Modeling
5. Integration with Building Design Process

Few BPS tools are recognized as design support tools to the same extent as design tools are. The functionality of BPS tools transcends the knowledge and skills base of only one discipline (Attia et al., 2012), namely the discipline of engineering. In brief, the results of Attia et al. show a common pattern that indicates a wide gap between how architects and engineers use BPS tools. One may argue that historically the only objective of BPS tool development has been to ensure accuracy. As a consequence, the models (the solvers implemented in BPS tools) have become increasingly complex, especially with the advent of cheap and powerful computing power (Ahsrae, 2005).

This choice typically depends on the use or type of the building, the complexity of the building and/or systems, and the application. The latter includes energy performance requirement, energy performance certificate and recommended energy performance measures. In brief, the choice between the two main methods is about the need to maintain a balance between accuracy, transparency, robustness, reproducibility (ISO, 2008), speed, usability, interoperability and integration (Attia et al., 2012). Proper coordination requires a dynamic view of all design activities, verification of their interrelatedness and anticipation of expected downstream impacts of alternative decisions.

A study from 2001 (Wilde et al., 2001) stated that BPS tools were rarely the course of subsequent energy efficiency improvements. Seen in this light, the methods and regulations applied to the design process may be of higher effect if the aim is to improve energy consumption and indoor environment. With the aim to better implement future tools into the design process, Wilde et al. (2001) thus identified the developments of design tools that may provide better support for building performance in the early design stage. Three major developments were highlighted (Wilde et al., 2001):

- Simulation tools for non-specialists (designer-friendly analysis tools)
- Better communication between architects CAD (design) tools and consultants and their (BPS) tools.
- Developments in integrated analysis platforms (Design-integrated tools)

Here it is evident that tools and methods are inseparable and the communication between tools is a challenging task. If simulation tools for non-specialists are to be used, it allows for early design performance evaluations without simulationists. Tools that can facilitate data between building designers' tools and a simulationist tool and integrated platforms (also known as design-integrated tools) may require very clear to achieve the facilitation.

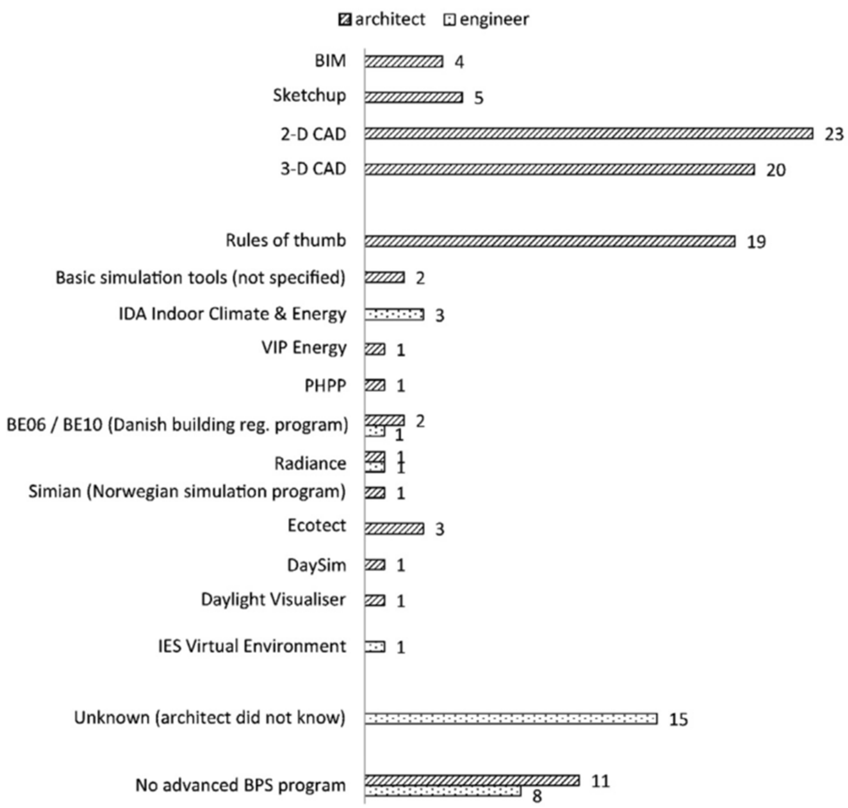


Figure 17 BPS tools used by Nordic architects in the design process, from Kanters et al. (Kanters et al., 2014)

In a more recent survey (Kanters et al., 2014) specifically focusing on architects' use of BPS tools to inform solar design[12] (see Figure 17), the architects in the Scandinavian countries (Denmark, Norway and Sweden) placed simulations of "absolute energy performance" in the hands of consulting engineers who had the legal responsibility for the outcomes. This fits the observations made previously by Augenbroe (2001). However, the Danish architects were the only building designers who were using advanced BPS tools themselves. Often, the building designers used the BPS tools as "real design tools by comparing different design alternatives relatively". Primarily, the use of the tools was to define the direction of the design and not the absolute performance. Most building designers, however, used rules of thumbs and/or consulting specialists (simulationists) to inform their design. Kanters et al. (2014) noticed that those building designers who used BPS tools often would use more than one tool at the same time during the design process to simulate several aspects of the energy performance of a

[12] Solar design is often associated with passive energy saving strategies and is considered an important part of High-Performance Buildings.

building, e.g. daylight conditions, energy production of solar technologies, and thermal balance. *Combining all such separate tools into one environment, using the same geometry model, would be preferable and speed up the iterative design process*. (Kanters et al., 2014)

In every project there are distinct stages calling for different types of assessments to assist design evolution and it is not possible to generalize the process through methods and tools. As established above, early stage assessments are mostly based on expertise and experiential knowledge of consultants and the need to keep this knowledge present in some way or another is still important (Augenbroe, 2001). Today, developments in providing design integration with BPS tools areas are still predominant with a shift towards the communication between tools and integrated platforms. Some of the current developments may be mentioned: RenewBIM (Gupta, 2013; Gupta et al., 2013) and Simergy (LBNL, 2014), OpenStudio (NREL U.S. Department of Energy, 2014), IESVE (Integrated Environmental Solutions, 2013), Sustain (Greenberg et al., 2013), Virtual Design Studio (VDS) (Michael Pelken et al., 2013; Zhang et al., 2013), and ZEBO (Attia, 2012). What is central for many of these developments is the aim to put an end to data loss in design. Bazjanac and Kiviniemi (2007) reasons that *the exchange and/or sharing of data among software applications have traditionally been disorganized and inefficient.* But these developments have improved data exchange quite radically during the past few years.

Data exchange

When possible, data exchange is usually based on “point-to-point” exchange via software interfaces that map parts of internal data structure and sets of one application to the other. Furthermore, data exchange associated with performance analyses in the early design stage is mainly kept in one direction from the design tool to the performance evaluation tool. Nonetheless, performance analysis in buildings is an iterative and dynamic process and the building industry is in most cases approaching building design as an open problem. This means that no method may cover the performance definition, integration and evaluation during the whole life-cycle process for every individual project. What is certain and true for all projects is that the building design is evaluated based on how it complies with set requirements. Those requirements can be prescriptive or performance based[13], see also Section 2.1.2.

Today, data exchange is often associated with the data exchanges to, from and between building information models (BIM). The two dominant data exchange formats are IFC (See et al., 2012) and gbXML (gbXML.org, 2014). More on these formats are found in Section 2.4.5, but what should be mentioned here, is that data exchange is much more than a file format. For example data exchange is the ability to link multiple software tools at run-time in order to co-operatively exchange information.

[13] Performance based requirements are also referred to as functional or objective-based requirements.

2.4.3 Data-driven approach - the non-BPS tools

The main advantage of the Forward approach is that the system does not need to be physically built to predict its behavior. Thus, this approach is ideal in the early design stage. It is straightforward to assume that the Forward approach is best suited in the evaluation of early design stage, hence the data-driven approach is ill fitted for early design stage analysis. However, it is not entirely the case. It is necessary to establish few references into the Data-driven approach. The reason is found in the importance of responsive feedback, which is discussed in Chapter 4. Data-driven methods for energy-use evaluation in buildings can be classified into three approaches, according to ASHRAE (Ahsrae, 2005):

- empirical or "black-box" approach
- calibrated simulation approach
- "gray-box" approach

Ahsrae (2005) explain the most common techniques in the data-driven approach is the use linear or change-point linear regression to correlate energy use, (or peak demand), with weather data and other independent input variables. These methods along with simple, or multivariate regression models, usually depend on measured building data.

Table 2 Typical Data-driven modeling methods from (Ahsrae, 2005)

Table 10 Capabilities of Different Forward and Data-Driven Modeling Methods

Methods	Use[a]	Difficulty	Time Scale[b]	Calc. Time	Variables[c]	Accuracy
Simple linear regression	ES	Simple	D, M	Very fast	*T*	Low
Multiple linear regression	D, ES	Simple	D, M	Fast	*T, H, S, W, t*	Medium
ASHRAE bin method and data-driven bin method	ES	Moderate	H	Fast	*T*	Medium
Change-point models	D, ES	Simple	H, D, M	Fast	*T*	Medium
ASHRAE TC 4.7 modified bin method	ES, DE	Moderate	H	Medium	*T, S, tm*	Medium
Artificial neural networks	D, ES, C	Complex	S, H	Fast	*T, H, S, W, t, tm*	High
Thermal network	D, ES, C	Complex	S, H	Fast	*T, S, tm*	High
Fourier series analysis	D, ES, C	Moderate	S, H	Medium	*T, H, S, W, t, tm*	High
ARMA model	D, ES, C	Moderate	S, H	Medium	*T, H, S, W, t, tm*	High
Modal analysis	D, ES, C	Complex	S, H	Medium	*T, H, S, W, t, tm*	High
Differential equation	D, ES, C	Complex	S, H	Fast	*T, H, S, W, t, tm*	High
Computer simulation (component-based)	D, ES, C, DE	Very complex	S, H	Slow	*T, H, S, W, t, tm*	Medium
(fixed schematic)	D, ES, DE	Very complex	H	Slow	*T, H, S, W, t, tm*	Medium
Computer emulation	D, C	Very complex	S, H	Very slow	*T, H, S, W, t, tm*	High

Notes:
[a]Use shown includes diagnostics (D), energy savings calculations (ES), design (DE), and control (C).
[b]Time scales shown are hourly (H), daily (D), monthly (M), and subhourly (S).
[c]Variables include temperature (*T*), humidity (*H*), solar (*S*), wind (*W*), time (*t*), and thermal mass (*tm*).

Black-box approach

For design purposes, the operation between measured energy use and the various influential parameters (e.g., climatic variables, building operation, and geometrical changes) existing measured data might not be sufficient as a data source for prediction purposes. Nonetheless, the use of machine learning algorithms e.g. Fourier series and Artificial Neural Networks (ANNs), allows for a prediction into a larger solution space from a limited parameter space of empirical data. But these statistical approaches are according to Corgnati et al. (2013) probably more suitable to evaluate demand side management tools, identify energy conservation measures in an existing building *and to develop a baseline model in energy conservation measurement and verification tools* (Corgnati et al., 2013).

Calibrated simulation approaches

Calibrated simulation approaches implies calibration with measured data; these types of approaches are reserved for monitoring and management purposes for e.g. energy savings in retrofit buildings; they are not directly fitted for building design.

Gray-box approach

The gray-box approach first formulates a physical model to represent the structure or physical configuration of the building or the energy system, and then identifies the representative parameters and aggregated physical parameters and characteristics by statistical analysis (Rabl and Rialhe, 1992). This approach also includes inverse models (steady state inverse models and dynamic inverse models). A model is dynamic when dependent or independent variables are explicitly expressed as functions of time. The criterion on which the classification is based is that dynamic inverse models contain time-lagged variables (Corgnati et al., 2013). Dynamic inverse models may include equivalent thermal network analysis, ARMA models, Fourier series models and machine learning algorithms. The dynamic models are capable of taking into account dynamic effects such as thermal mass, which traditionally has required the solution of a set of differential equations.

2.4.4 Integration between BPS tools

Research which is dedicated to improving the integration of multi domain simulations often seek to make it easier to consider different performance aspects (comfort, health, productivity, energy, etc.) at different levels of resolution in terms of time and space (region, town, district, building, construction element, etc.). The whole point of integration is that multi domain BPS tools can support information exchange throughout a simulation. It is important to establish these integration methods in relation to the integration mechanisms within integrated dynamic models.

Hensen (2004) has categorized the four main developments in integration between building performance simulation environments.

- Data and process model integration
- Data model interoperation
- Process model interoperation
- Data model and process model co-operation

The most widely used approach, the data and process model integration, is based on providing a facility to simulate different sub-domains within the same program.

This approach is also known when combined with a design tool as a *combined model method*, as discussed in Section 2.4.5. There are many research projects in this area; for example some of the most widespread tools among building designers in the Scandinavian countries (Kanters et al., 2014): Ecotect (Autodesk, 2013b), IDA (Equa, 2014), IESVE (Integrated Environmental Solutions, 2013). From a user point of view, the main disadvantage of this approach is that the user is still restricted to the options / features offered by a particular environment or program, which is developed by a single research unit or a small group of researchers. What is emphasized by Hensen (2004) is that the latter does not make it very attractive for other researchers to join in at a later stage and it does not really enable shared developments.

In rough terms, Hensen (2004) distinguishes between other approaches by their ability to link applications either at run-time, through modeling language or through file formats in order to co-operatively exchange information. To achieve the highest form of integration the run-time and modeling language are to be preferred. But as Zhai (2003) emphasizes, the different methods all have eligibility. Zhai (2003) prefers to categorize the coupling mechanisms between simulation tools by various coupling strategies as illustrated in Figure 18. In brief, the static couplings are described as pre-calculated results from one tool then read into the other tool for further development. The dynamic couplings are reserved for two (or more) BPS tools interacting with one another, while the bin methods are a compromise between the dynamic and the static method in view of reducing computational costs.

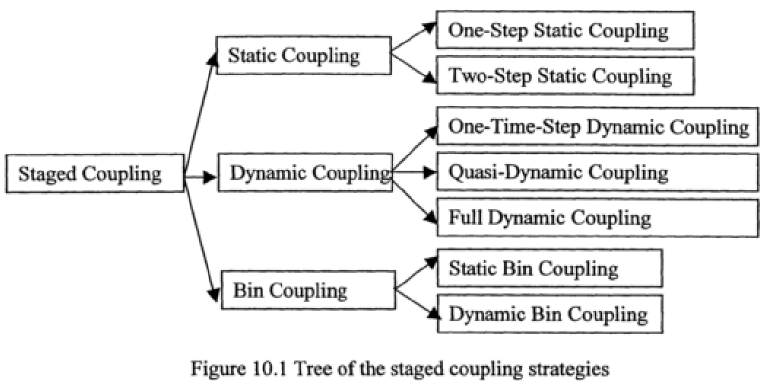


Figure 18 Tree of the staged coupling strategies between the MIT-CFD tool and EnergyPlus. From Figure 10.1 Zhai (Zhai, 2003)

Citherlet et al. (2001) discussed multi domain model integration (again focusing on dedicated BPS tools) and argued that integrated applications may ideally take data and computer models (solvers) from multiple sources and process this through a unified integrated tool. While the researchers do not discuss the integration of dedicated design domain they argue that only two ideal ways to go about this multi domain (performance) integration exist. The coupled approach is able to link multiple solvers at run-time in order to co-operatively exchange information as shown to the left in Figure 19. This approach, which basically does the same as “the integrated approach” (to the right in Figure 19), has one disadvantage: The coupled approach lacks data and link consistency between the coupled solvers, where the integrated approach has the advantage of a unified program interface between solvers. The ideal application would allow design as a domain and any performance evaluation domains to be integrated by any of these two approaches.

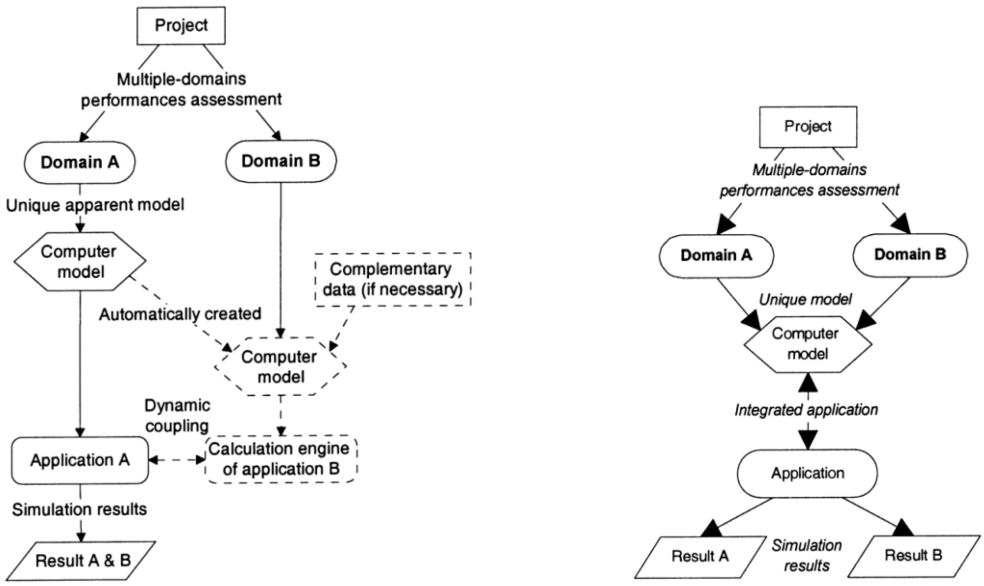


Figure 19 Multi domain coupled approaches operated through the unified application, to the right: Coupled Approach, to the left: Integrated Approach (Citherlet et al., 2001)

Citherlet et al. (2001) argues that the application of computer-supported design environments has to benefit from simulation tools during the design process. To do this the disadvantages of completely separated tools can be avoided if are BPS tools integrated with design tools. However, the integration of BPS tool to another BPS tool as discussed above cannot be fully compared to the integration of design tools and BPS tools.

2.4.5 Integration of design tools and BPS tools

Processing a geometric model from design software to a BPS environment has often been associated with a manual export/import task, and most BPS tools today only support the unidirectional approach from design tool to BPS tool. Much research over the years has tried to better integrate design and performance tools. While there are many reasons to couple the two categories of tools, this thesis will follow the ends defined by Clarke (2001) aiming to create a *computer-supported design environment* that is able to *automatically access the data describing the design and give feedback on all aspects of performance and cost in terms meaningful to the building designer*. (Clarke, 2001)

The SEMPER projects (Lam et al., 2004; Mahdavi et al., 1997) were some of the first large scale projects that managed to integrate the design tool and the BPS tool based on *dynamic links* (Mahdavi et al., 1997). As most research projects seeking to define a multi-aspect design environment to different BPS tools, the SEMPER projects were proof of concepts. The one key objective that distinguishes SEMPER from e.g. Zhai's (2003) integration approach is the ambition to create:

> *"'Seamless and dynamic communication between the simulation models and the architectural design representation"* (Mahdavi et al., 1997)

Still, Mahdavi et al. (1997) notes that SEMPER does not directly deal with complex configurational aspects of buildings, *such as form and massing*. In this light, the need to conduct performance evaluations of project specific geometry models is regarded as a highly complicated task.

Recent papers (Bazjanac, 2004; Sanguinetti et al., 2012; Shi and Yang, 2013; Thuesen et al., 2010; Zarzycki, 2010) examine how a geometric model dynamically can be operated in relation to building performance simulations. Some approaches seek to unify the design tool and the BPS tool by defining a common exchange format (data model operation), while others operate the geometrical model in the very same environment that facilitates the building performance simulations (data and process model integration). Yet others use a middleware to facilitate the coupling between the dedicated design tool and the BPS tool. The basic shared idea is to combine the benefits of dedicated design tools and dedicated BPS tools. Some of these approaches may be compared to the multi domain coupled approach illustrated in Figure 19, but only one of the two domains is now represented as a design tool. If run-time linked, fully integrated and controlled by a unified environment, the approach may be compared to the dynamic couplings shown in Figure 18. Or if the two tools are merged they may be comparable to the integrated approach illustrated in Figure 19, on the right hand side. Either way, they form an integrated computer-supported design environment as requested by Clarke (2001).

There are three methods (Negendahl, 2015a) to integrate design tools and BPS in the early design stages (see Figure 20).

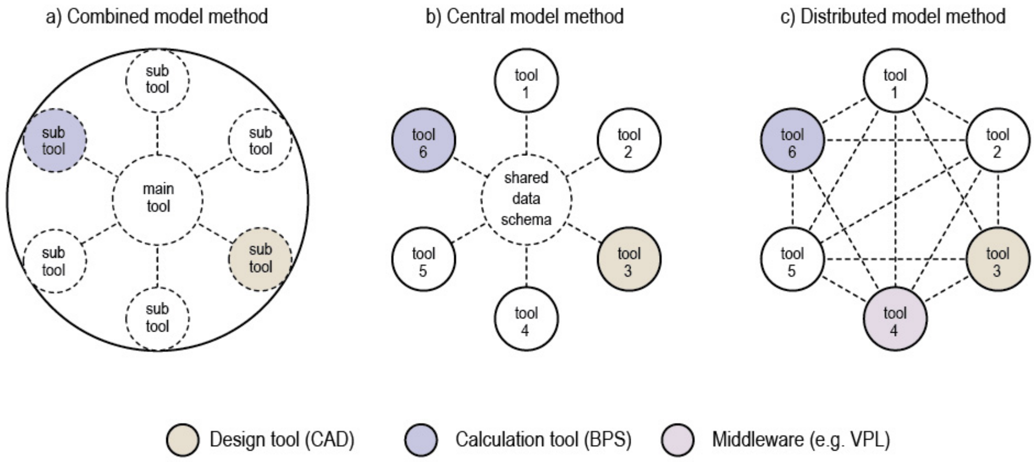


Figure 20 Differences between coupling methods: a) Combined model method (typically operated in a simulation package), b) Central model method (using a central database/file format/schema), c) Distributed model method (utilizing a middleware). (Negendahl, 2015b)

Combined model

Simulation packages, e.g. IESVE (Integrated Environmental Solutions, 2013), contain a *combined model* (Figure 21) and have the advantage of the operator being able to control the precision of the model within all steps of model production, manipulation and simulation. The combined model can handle both the modeling and the simulations at run-time level and provide consistency of the environment, which is an attractive feature for many users. The clear advantage of a combined model is that the design tool functionalities and the simulation tool functionalities essentially are integrated, thus enabling *tool domain integration* as discussed in (Negendahl, 2015a). The main disadvantage of this method is that the user is restricted to the options and features offered by a particular environment or program (Hensen, 2004).

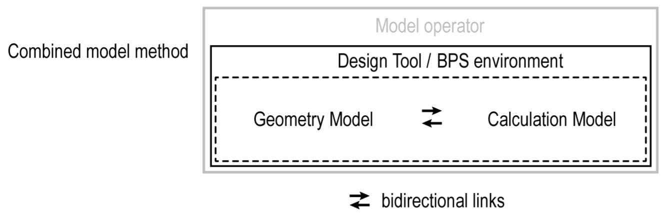


Figure 21 Combined model. Consists of a design tool and BPS in the same environment. (Negendahl, 2015b)

While combined models are limited to the functionalities of the modeling environment, most combined models support import and export geometry from other tools, but do not support dynamics of bi-directional updates between external tools. The concern of using simulation packages and combined models is that all involved participants must agree to use the same principal tool to maintain the high convergence between models. In this regard, it may be difficult to use combined calculation models in larger uncoordinated groups and loose interdisciplinary projects.

Central model

The *central model* (Figure 22) is based on a widely used central framework. The concept of centralizing building information data in a shared data schema is typically associated with the early influences of a buildingSmart initiative (BuildingSmart.org, 2013). Various tools read and write to the same model and are thereby able to connect semantic information from a design tool to a BPS environment. Of recent methods based on IFC as a coupling medium, RenewBIM (Gupta, 2013; Gupta et al., 2013) and Simergy (LBNL, 2014), and for gbXML as a coupling medium, OpenStudio (NREL U.S. Department of Energy, 2014), can be mentioned. Since design tools and BPS tools only recently have begun exchanging data through these types of open formats, there seem to be no consensus whether the design tool or the BPS tool should handle the convergence. Rose and Bazjanac (Rose and Bazjanac, 2013) suggested to use an intermediate algorithm to create IFC-compliant space boundaries from a geometrical (CAD) model, thereby assisting the process of creating BPS friendly geometry. The implementation of other automated or semi-automated algorithms, such as automatic thermal zoning or simplification of sophisticated building geometry, will be needed in either design tools, in BPS tools or in the IFC schema itself to make the coupling process between design tools and BPS tools automated.

The geometric model and the calculation model can in theory be *dynamically coupled* (bi-directional linked on run-time level) with an exchange file format, however this is rarely the case with the central modeling method used today. The main idea is to unify the design tool and the BPS tool by defining a common exchange format. Most frequently, the tools using a central model are capable of exchanging data with other dedicated software environments and are typically based on IFC or gbXML. Among these coupled or linked tools are BPS environments that are devoted to making energy and indoor environment performance calculations.

Essentially, the geometric model and the calculation model *live* in a shared format, and every tool that supports the format is able to operate in the model. Implementations such as DCOM (Amor and Faraj, 2001) are characterized by the reliance of a common data schema as the interoperability gateway between software, and are therefore considered a central model method.

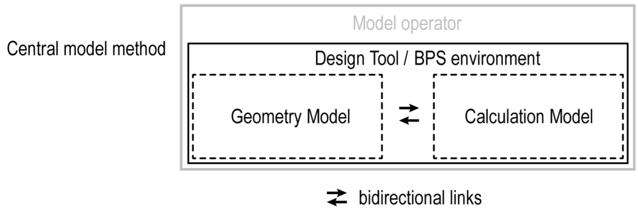


Figure 22 Central model: a combination of a design tool and a BPS environment. (Negendahl, 2015b)

Moosberger (2009) has shown examples of how unidirectional IFC couplings to the BPS IDA ICE (Equa, 2014) can improve simulation convergence between the geometric model and the calculation model. This unidirectional method is widespread and does little to help provide the building designer with relevant performance feedback, unless the building designer is the operator of the calculation model. Plume and Mitchell (2007) attempted to utilize the fully centralized idea to perform different performance simulations. While IFC is capable of containing most of the data needed for the various BPS tools, the building model needs to be constructed with collaborative interchange in mind and capable of *anticipating the needs of design collaborators* (Plume and Mitchell, 2007). The central model has to be operated in consensus with all involved parties. As a consequence, collaboration within centralized models has been considered time consuming and in some cases counterproductive in terms of design exploration. Using common data schema like IFC and gbXML to structure information exchange, regardless of whether it is proprietary or an open standard, imposes restrictions on how designs can be described and thus explored (Toth et al., 2012). Models translated from a shared building information model[14] are as precise as the database or schema allows it to be. Limitations of the read and write structure derive from poor data quality of a single object, which will agitate through all of the connected environments of the database/schema. The main problem, however, is presently not the open file formats but the lack of software support and user support of common open file formats.

The possibility for a simulationist to obtain the useful information needed for analysis directly from an architectural model is currently limited. According to the CEO of Graphisoft Várkonyi (2010) is the central model in practice decomposed into several models. Because, a single model is practically unable to serve the requirements of each discipline in the design team. Thereby, each discipline operates its own model with regular synchronization of the changes with the other models, using a common “reference model” (Várkonyi, 2010). This reference model acts as a central master model, and any changes synchronized with this will be applied to every other sub model. However, the automatic updating of data when changes are made to one model is not always allowed. Therefore, manual

[14] Referring to the two dominant implementations of BIM-standards on the market, gbXML and IFC

corrections and manual exports/imports are often handled within these central model methods. Most design and BPS tools have implemented the option to import IFC as well as ‘Save as’ or ‘Export’ the model, in addition to their proprietary data formats (Eastman et al., 2011). However, these options are insufficient for an effective exchange, because IFC can be ambiguous due to the fact that it offers several ways to define objects, relations and attributes. Therefore, IFC implementations require a clear guidance for specific purposes and projects.

To better guide software developers in bringing uniformity to IFC data exchange between disciplines, various specifications have been suggested. These specifications are called Model View Definitions (MVDs). They identify what should be expected from an IFC (Eastman et al., 2011), while they document the way data exchanges are applied among different applications. MVDs are defined by buildingSMART (BuildingSmart.org, 2013) as “a subset of the IFC schema that is needed to satisfy one or many Exchange Requirements of the AEC industry”. Eastman et al. (2011) argues that thanks to MVD the “explorer knows what is required and the receiver knows its content”, and therefore the gap between the export and import of data is reduced. Another way to improve data exchange between disciplinary specialist sub-models is through Information Delivery Manuals (IDMs). While MVD is aiming at mapping exchange requirements to IFC, IDM is aiming to capture processes and exchange requirements of the schema (Karlshøj, et al. 2012). Finally, a supplementary exchange schema named BIM Collaboration Format (BCF) has been widely employed in unison with IFC. BCF allows only the “relevant issues”, and not the entire BIM, to be exchanged between software packages (Granholm, 2012). BCF is now implemented in Tekla Structures, Solibri Model Checker and DDS Architecture, and claims to improve the workflow and reduce the transfer of large BIM files. Recently, buildingSMART acquired the ownership and the rights of the BCF schema to adopt and keep it as an open standard (BuildingSmart.org, 2013).

Another method that may be classified as a central model method is OpenStudio (NREL U.S. Department of Energy, 2014) using SketchUp as a design tool and coupling the BPS Energy+ by its own file formats. Kalay (2001) suggested a method often referred to as *integrated collaborative design environments* (ICDE’s). The concept ranges beyond the interoperability with BPS through a central model when suggesting the inclusion of *Evaluation tools, Negotiation tools* and *semantically rich databases.* Elaborate semantic systems, as described by Kalay (2001), have been developed in different prototype forms over the past decade. An example is suggested by Wurzer (2010), who successfully aided building designers in automating process-planning; nevertheless no commercial product has yet been developed and used in practice.

Distributed model

Nederveen et al. (2013) argues that a distributed model fits better with the distributed responsibility and role definitions that are found in most collaborative practices.

BIM environments are usually visualized as a circle of actors (disciplines, applications) positioned around a central Building Information Model. However, more and more people involved in BIM state that there is a need for a more decentralized approach. Not everyone believes in the ideal of a “central BIM” anymore, (Nederveen et al., 2013).

Distributed model methods (Figure 23) can be seen as a response to the central model concept, disengaging themselves from a top-down control and one directional model operation. While it is important to note that *architects’ definitions of buildings* in design tools *do not necessarily reflect the needs of energy simulation* (Bazjanac, 2004), the tools may need to be able to adjust, conform, enhance and even eliminate parts of the model to be successfully interpreted by a BPS tool. This might be the largest problem of using a central model, as the model framework is depended on placeholder content that might never be created. Decentralizing the effort of creating content, however, will help model convergence. Nevertheless, decentralizing efforts of modeling may in worst the case scenario end up returning to the state of complete incoordination at model level. Because of this, distributed models are characterized by deep integration at model level by utilizing a middleware component to translate data between the design tool and BPS tool.

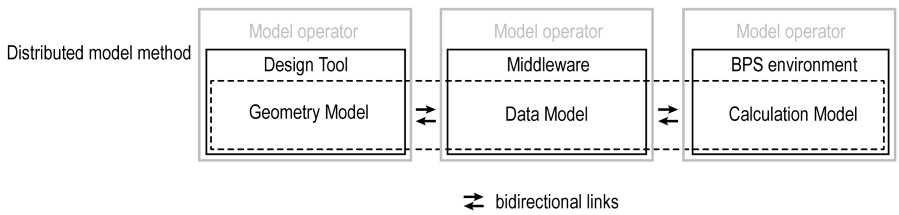


Figure 23 Distributed model: a combination of a design tool, a middleware tool and a BPS environment. (Negendahl, 2015b)

Bazjanac (2004) demonstrated that IFC is capable of providing sufficient interoperability in coupling Energy+ as a BPS, but a *middleware element* was necessary to exchange data. Additionally, he found that it was necessary to input secondary data required for the HVAC system, such as occupancy and use schedules. Since Bazjanac’s investigations of IFC, various improvements and extensions have been added by the buildingSMART initiative. IFC2x4 now supports many of the then missing schedules and other technicalities missed in the interoperability investigations. Nonetheless, the improvements in the schemas do not change the fact that data is not generated by operators, and if data is generated, it may not be stored in the schemas in a way enabling other tools to understand and use it. The middleware is therefore not merely a simple converter between formats and platforms, but a system that is able to filter, modify and extend operator definitions to such a degree that the definitions reflect the needs of BPS tools (and obviously the needs of the design tools).

One of the recent developments in distributed model methods is Sustain (Greenberg et al., 2013), Virtual Design Studio (VDS) (Michael Pelken et al., 2013; Zhang et al., 2013), and ZEBO (Attia, 2012). Geometry models from design tools such as Revit (Autodesk, 2013c), Rhino (Robert McNeel & Associates, 2013a), and SketchUp (Trimble, 2013) are imported through the middleware into the BPS. VDS, for example, functions as the necessary middleware to distribute and modify data to BPS tools, such as CHAMPS (Feng et al., 2013) and Energy+ (U.S. Department of Energy, 2013). VDS seeks to provide a hybrid tool made of components from architectural design practice and engineering simulation techniques and relates itself to existing assessment systems such as LEED, DGNB etc. At the same time it incorporates advanced automation features, such as optimization algorithms and auto generation of e.g. VAV systems. While VDS supports various design tools, the actual model control and feedback is provided in the VDS system itself and as a result all operators must have direct access to the system to react upon the feedback.

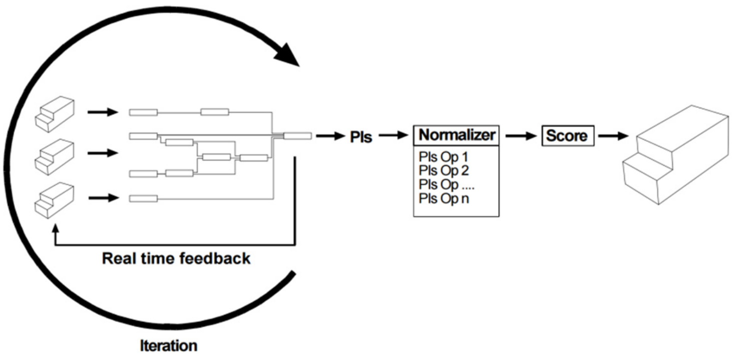


Figure 24 Real-time feedback and selection process, from (Sanguinetti et al., 2010)

Sanguinetti et al. (2010) argued that the complexity and iterative nature of the generative synthesis process (the process of a designer to perceive and interact with e.g. a tool see Figure 24) renders the integration of building performance parameters dependent on the interoperability between design and BPS tools. This typically delays the analysis step until after a design solution is fully developed, which means real-time analysis of design synthesis cannot be fitted into the process. Their solution was an integration of design synthesis and analysis centered around a design tool. This was implemented in the design tool

Rhino and rather than utilizing in a typical BPS tool, Sanguinetti et al. (2010) used RhinoScript to facilitate the analysis feedback from normative calculations in spreadsheets.

In this way building designers may receive real-time feedback from the performance indicators, in a two-way process. When the designer has defined a parametric model it is fully controlled by this external spreadsheet. Hence, for every change in the parameters, actualization occurs in the parametric model as well in the spreadsheet, providing real-time feedback to the user.
The main advantage of their approach is the synchronization between parametric modeling and/or scripting environment and performance based calculations, providing feedback on geometric and material variation. The parametric approach enables direct visualization of the effect of the calculation, without intermediate steps of design modifications. The weakness of this approach is due to the limitations of the tools to support complex mathematical calculations, i.e. the spreadsheet calculation. In addition, this approach requires an expert user to define the architecture of the design model, including parameters, rules, and constraints (Sanguinetti et al., 2010). The method was found to be highly flexible and could serve project specific design explorations including almost any qualitative consideration; however, the use of spreadsheet calculation does create concerns of tool validity (Negendahl and Nielsen, 2015).

Another example of a parametric middleware is suggested by (Nembrini et al., 2014), heavily depending on automatic model translations between various simulation tools. Additionally, the authors demonstrate the advantages of implementing a scripting language interface. And like Sanguinetti et al. (2010) they argued that some of the advantages between parametric scripting and BPS tools are:

- Easy investigations of design variations
- Fast and easy tuning of model complexity, often needed in the early design stage
- Systematic exploration of the defined parameter space

The main drawback in relation to the approach of Nembrini et al. (2014) is their implementation for geometry modifications; it is dependent on the object-oriented language Java, which is rarely used by building designers and simulationists.

As the middleware is an essential part of a distributed model, the flexibility, features, and usability of the middleware is key to how the model interoperability converges. An *Integrated dynamic model* as illustrated in Figure 25 is a distributed model, where the middleware consists of a visual programming language (VPL).

The integrated dynamic model

In a recent review, Negendahl (2015) defines an integrated dynamic model as a special case of a distributed model. An integrated dynamic model is a combined model composed of a geometric model controlled in a design tool dynamically coupled to a visual programming language (VPL), which is again dynamically coupled to a building performance simulation (BPS) environment. The middleware can be operated by either the simulationist (Bleil de Souza, 2012), or the building designer, both of them or by a third, undefined operator.

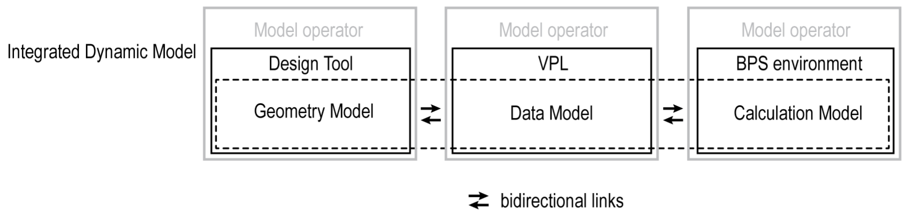


Figure 25 Integrated dynamic model: a combination of a design tool, a VPL (visual programming language) and a BPS environment. (Negendahl, 2015b)

VPL's such as Grasshopper (Robert McNeel & Associates, 2013b), Dynamo (Autodesk, 2013a), GenerativeComponents (Bentley, 2013), Digital Project (Gehry Technologies, 2013), and Yeti (Davis, 2013b) are examples of some of the *scripting tools,* designers and architects are using to automate form generation. Arguably, VPLs are able maintain the design variables as open and parametric, and code instructions are more user-friendly than those provided by lower level programming languages such as Java, RhinoScript, etc.
As the VPLs are run-time coupled to the design tool, the coupling can be defined as *dynamic* in the way Zhai (2003) categorizes the couplings between BPS tools. Also the integrated dynamic model method can be used for multi domain (performance) evaluations, hence the model can be categorized as *integrated,* much like Citherlet et al. (2001) categorizes multi domain BPS tool couplings.

VPLs can in some cases be considered as design tools themselves, mainly because of the heavy use of geometric modeling functionalities. The reason why these tools are categorized differently than traditional CAD tools, is their ability to handle non-geometric data, and let operators create their own algorithms (Negendahl, 2015a). The VPL is coupled bi-directionally with one or more design tools, e.g. Rhino, Revit and MicroStation (Bentley, 2014) and has direct run-time access to the design tool functions. VPLs coupled to design tools are able to formalize to the exchange of data consisting of collections of geometric primitives, and the *geometric-content-based data exchange* of a VPL is *in opposition to BIM's 'assigned-attribute-based' data structures* (Davis and Peters, 2013). This means that VPLs facilitate data across any content and object relationship, while schemas like IFC prescribe object relationships through attribute data. The ability

to cross-reference any relationships (both geometrical and non-geometrical relationships) provides a highly flexible and open environment. However, the presence of VPL does not in itself guarantee interoperability between compatible software and dependencies, and rules of transferring data between tools must be defined in every integrated dynamic model (Negendahl, 2015a).
In contrast to the model suggested by Nembrini et al. (2014), which also utilizes a scripting language the integrated dynamic model has a dedicated design tool under the hood, and the scripting tool is defined as a higher level language. These two differences make the integrated dynamic model more approachable for building designers with limited coding experience.
Some of the recent frameworks that support the dynamic couplings between design tools, visual programming languages, and BPS are GenerativeComponents combined with Design Link (Holzer, 2010; Holzer et al., 2009), which again delivers run-time couplings to Energy+ and Ecotect (Autodesk, 2013b). DEEPA (Toth et al., 2011) is one of the more recent attempts utilizing the VPL GenerativeComponents in combination with Energy+, in this case by using IFC to maintain import/export compatibility towards other coupling directions. Green Building Studio (Bambardekar and Poerschke, 2009; Lin et al., 2013) is normally classified as a centralized calculation model, but since the framework is able to use the design tools Revit and Vasari (Autodesk, 2013d) through the VPL Dynamo, together the tools can form an integrated dynamic model. The Rhino-Grasshopper-combination supports wide a range of couplings to various BPS. The facilitation of links to the BPS is handled by third party modules, such as Viper (Solema, 2013) (Energy+), ArchSim (Dogan, 2014) (TRNSYS (Thermal Energy System Specialists, 2013)), Geco (UTO, 2013) (Ecotect/Radiance), DIVA (Jakubiec and Reinhart, 2011; Solema, 2013) (DAYSIM (Daysim.ning.com, 2013)) and Honeybee (Roudsari et al., 2014) (Energy+). All of these modules are coupled through the VPL Grasshopper to the design tool Rhino.

Davis (2013a) has compared the paradigms and scope of the most commonly used VPLs in architecture listing some of the features and limitations of the different programming languages. Davis' main concerns are not tool-to-tool interoperability and performance related feedback, but focus on user-to-tool support, as well as how the VPLs are built and applied in the architectural processes. As he describes, many building designers have embraced the new (VPL) tools, and examples of extensive utilization to generate form with VPLs are a trending design strategy among building designers. Burry's (2013) main argument for using VPLs in his book *Scripting Cultures* is *the potential to free up the designer to spend more time on design thinking.* Scripting and the use of VPLs are often associated with parametric automation of geometry in architecture, but geometry is just one aspect that VPLs are able to automate. Some of the more advanced automations are discussed in (Negendahl, 2015a). As long as the VPL is able to interpret the data, the VPL will not distinguish between data types. Toth et al. (2012) sees the *elimination* of the common data schema as a prerequisite for information exchange, allowing design freedom to create custom digital workflows unfettered by standardization constraints while distancing itself from the central model method. Toth et al. (2012) further argues that the VPL does open up for exchanging data with IFC and other open formats, and the VPL can so to speak *act as a gateway* to a common data schema.

Performance simulation tools allow the user to parametrically analyze and predict the complex interaction-patterns involving these variables. This can be linked to the view that simulation should not be used only for final performance confirmation but as an integrated element of the design process (Augenbroe, 1992). Also, over a decade ago Mcelroy & Clarke (1999) pointed out that the simulations can be performed within the design practice by those that design.

2.5 Conclusion

It can be concluded that there is a consensus amongst researchers that decisions in the early design stage have large impact on the final design outcome. Therefore early design decisioning need early design stage performance evaluation. Thus design and performance evaluations can be positively assisted by BPS tools and other data driven approaches. Design methods such as the Integrated design process and the use of building (pre)certification systems that precepts BPS tools have shown theoretically to improve the likelihood of creating High-Performance Buildings.

However, if design *and* performance are to be considered in parallel, the available BPS tools to support the above methods need to be integrated into the design process in new ways. No single tool manage to gap the design thinking and flexibility needed for a building designer, while it can simulate and create meaningful performance feedback on building performance related issues needed of a simulationist. Today early building performance simulations are quite common, and it is custom to use separated BPS models and design models. This segregates the performance of a building and the design of a building in an undesirable way as it further separates the meaning of *design* and *performance* rather than integrating the two. The need of simulation specialists (simulationists) to create, run and analyze the separated BPS models have increasingly intensified the need of design methods such as the Integrated design process and certification systems. These methods ensure the simulationist is able to interact with the building designer from the early design stage. However these methods do not ensure tool-level or even model-level interactions between the building designer and the simulationist.

To better reflect the modelling need of building designers and the model requirements of the simulationists a shared model has to be constructed. This model needs to bridge between a design tool (which are most likely to support the building designer) to one or several BPS tools that are most likely to generate the performance feedback. In turn this has potential to push the design alterations in the direction of High-Performance Buildings. It is concluded that faster couplings between the design tool and the BPS tools are needed and ideally run-time couplings delivering live feedback from the BPS tool need to be developed as few of these couplings exists.

The idea of including a VPL in the mix of tools (addition to the design tool and the BPS tool(s)) to construct a model with a number of open design variables allows the design team to set up parametric relationships on object level; it also allows rule based modeling procedures that ease the data transfer to the BPS tool and back again to the design tool. The VPL also introduces the whole concept of parametric, computational and generative modeling, which among other things can reduce the “cost of changes” and help the building designer to create geometries not possible by hand and conventional design tools. Conclusively, integrated dynamic models consisting of a combined model shared between a design tool, a VPL and one or several BPS tools can in many ways support the early design stage better than existing alternatives.

3. Optimization

Optimization covers a wide range of processes in building design from manual and heuristic attempts to full automation of optimization processes and workflows in large teams. To limit the discussion of optimization within the scope of the thesis objectives, optimization is defined as *computer automated building performance optimization in the early design stage*. This makes optimization a clearly non-heuristic process. However it does not make heuristic optimization less relevant for the early or later design stages. Heuristic approaches may as well be used in combination with integrated dynamic models; in fact the mere prospect of parametric modeling in combination with automated BPS tool analyses is the foundation of the motivation of Consequence based design (2.1.6). What is exclusive to computer automated optimization is the speed in which highly complex multivariate systems can be solved and according to Kataras (2010) the potential of finding better performing solutions in limited time frames is far more plausable than using heuristic processes.

Computer automated optimization implies the use of automation of simulations and generative processes such as machine generation of variations on geometry, materials, systems etc. In direct continuation of the previous chapter, optimization is discussed as a means to provide relevant building performance support to the design team in the early design stage. Relating to the conclusions of the previous chapter, the relevance of optimization revolves around ways to combine optimization with integrated dynamic models to support Consequence based design. The chapter concludes that integrated dynamic models can effortlessly host a wide range of optimization algorithms. Therefore, these models are able to optimize buildings in terms of the objectives of High-Performance Buildings. Even though integrated dynamic models may support optimization better than any alternative methods and tools today, the inclusion of qualitative objectives is still not a fully resolved task.

3.1 Introduction to early design stage optimization

> *"One of the problems with optimization is that not everything is captured by the fitness function"* (Davis et al., 2014)

Research to this day rarely account for the architectural design in the optimization process. Methods that handle non-quantifiable objectives in optimization are often heuristic based and as a consequence, optimization of building design that includes "architectural concerns" is a relatively unexplored research area. Consequently, the inclusion of optimization in the early design stage creates high bias towards quantitative (performance based) objectives. The reason can be found in the optimization methods available today where most research that seek to merge the optimization of building design with building performance, utilizes techniques that involve pre- and post-processing of the optimization process.

This includes the pre-process of clearly defining objectives and boundaries, and a post-process of selecting the *most suitable design alternatives* (Mora et al., 2008). As discussed in Section 2.1 problems within the early stages of building design involve the complex interaction between many quantitative and qualitative objectives. In optimization the objectives of interest can be very difficult to quantify and those that can be quantified may need very specific pre-processed rules applied only relevant for the particular project. Human evaluations as means to quantify an objective cannot (currently) be included in (computer automated) optimization. Thus any type of human intervention must either be included in pre-process or the post-process or both. Pre-processing objectives qualitative and quantitative and the interactions between these is not a simple task. Of this reason optimization is often applied to quantitative objectives alone and in most cases only one or few objectives are chosen for building design optimization.

This chapter explores and discusses how tools and methods include the optimization of qualitative and quantitative objectives in the early design stage. This discussion touches how “architectural concerns” can be maintained by “artistic control” and how multiple building performance related requirements can be handled by optimization algorithms. First the concept of optimization is discussed which include the principles of constraints and boundaries as well as methods to weigh multiple objectives. Secondly the way optimization is used in research and practice is discussed, here we touch upon the tools available today and the most popular optimization algorithms (search methods) present. Finally the most pressing challenges in using early building design optimization is discussed, these challenges are divided into methodical challenges and tool challenges.

3.1.1 Definition of optimization

Optimization is often explained as a (computer automated) process of changing user defined values (also known as vectors or attributes) of design variables x to find specific solutions, which are dependent on these variables. The key to understand here is that optimization can only work when the user has clearly defined what variables are relevant for the computer to alter and what dependencies/cost functions (often associated with objectives) the computer need to watch and aim for. In the early design stage, variables may be building volume, window geometry, façade type, etc., which are obviously of interest to the building designer. The design variables define a cost function $f(x)$ that needs to be minimized or maximized, and are most likely subject to various sorts of constraints.

Generally, an optimization problem can be represented in mathematical form: $\min_{x \in \boldsymbol{X}} f(x)$, where $x \in X$ is the vector of the design variables, $f: X \to \mathbb{R}$ is the cost or the objective function, and $X \subset \mathbb{R}^n$ is the constrain set.

In this chapter, cost functions defined as being objective-driven based on feedbacks from a BPS tool (section 2.4.2) or alternatively data-driven approaches (section 2.4.3). This includes the aim to optimize building performance in order to meet the objectives of High-Performance Buildings, e.g. to reduce (minimize) energy consumption and improve (maximize) indoor environment. Therefore cost functions do not explicitly include nor exclude qualitative objectives such as aesthetics, functionality, social sustainability, etc.

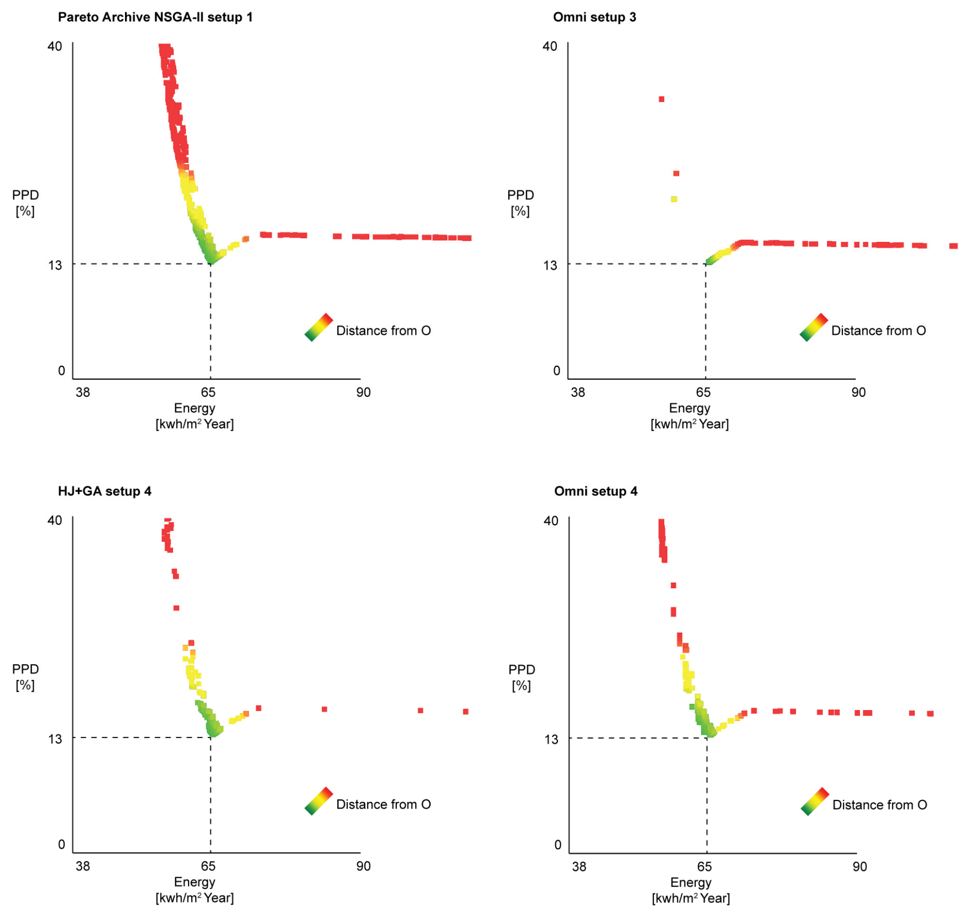


Figure 26 Pareto frontiers same optimization problem (energy consumption vs indoor environment, PPD) with four different approaches. All optimizations were here performed with MOBO (Palonen et al., 2013) with IDA ICE (Equa, 2014) as BPS tool.

Constraints

For some optimization problems it is necessary to impose constraints on the independent variables and/or the dependent variables. Especially related to multivariate optimization, constraint functions are useful to balance the multiple criteria. Constraint on a dependent variable is usually associated with a “penalty” imposed on one of the objective functions, if the objective or one of the other objectives is violated. For example, this can be implemented on every evaluation of a thermal comfort criterion (which is a dependent variable). When the criterion is violated, a large positive number can be added to e.g. the PPD (Fanger, 1973). Thus, if $ppd(x)$, with $ppd: \mathbb{R}^n \rightarrow \mathbb{R}$, denotes the predicted percent of dissatisfied people (as a percentage), and if we require that $ppd(x) \leq 10\%$, this could be implemented by the use of the inequality constraint $g(x) \triangleq ppd(x) - 10 \leq 0$ (Wetter, 2011a). This function could then be used instead of the original PPD evaluation. An example of this implementation was created by using MOBO (Palonen et al., 2013) for the BPS tool IDA ICE (Equa, 2014) and, as shown in Figure 26, Omni setup 3. Omni refers to the optimization algorithm Omni-Optimizer (Palonen et al., 2013). Omni setup 4 is implemented by multiplying the results of PPD and energy consumption as such: $f_{energy}(x) \cdot (f_{ppd}(x)/100)$.

Another common constraint method is the use of Barrier functions that impose a "punishment" with weighting factors on the cost function, if a dependent variable gets close to the boundary of the feasible region. According to Wetter (2011b) a drawback of Barrier functions is that the boundary of the feasible set can in many design problems be difficult to determine. By selecting small weighting factors, one can get close to the boundary. However, too small a weighting factor can cause problems for the optimization algorithm, as it will cause the cost function to be ill-conditioned. Penalty functions are probably the most widely used methods to balance out multiple criteria. Penalty functions allow crossing the boundary of the feasible set, and they allow implementation of equality constraints of the form $h(x) = 0$, for $\mathbb{R}^n \rightarrow \mathbb{R}^m$. The main difference between a Penalty function and a Barrier function is that cost functions are multiplied by a positive weighting factor μ, which is monotonically increased for penalty functions but monotonically decreased to zero for barrier functions (Wetter, 2011a). Thus, Penalty functions add a positive term to the cost function if a constraint is violated.

Pareto optimality

Utilizing stochastic optimizations (see Section 3.3 for details on stochastic optimization algorithms) in multivariate problems like building design, competing criteria are often unevenly balanced, and their relative importance is generally not definable. Therefore, instead of relying on Penalty functions and weighting factors of individual objectives to find optima, the use of non-dominated optimization methods can help the design team choosing solutions that are less biased towards singular objectives. These non-dominated optimization methods are often referred to as Pareto optimality[15] methods. And arguably, they help to identify a set of feasible designs that are equal-rank optimal. Pareto-ranked solutions (as shown in Figure 26) mean that no solution in the set is dominated by any other feasible design for all criteria (Grierson, 2008). Many researchers have utilized Pareto-ranked optimality methods in building design, e.g. Raphael (2011) , Wang et al. (2005), Stouffs et al. (2013) and Lin et al. (2013). Pareto optimality methods are usually part of a post-balancing process, where the non-dominated solutions are presented as a catalogue of candidates to the design

[15] Vilfredo Pareto (1848–1923) developed the concept known as 'Pareto optimality' of equilibrium positions, from which it is not possible to move so as to increase the utility of some entity without decreasing the utility of another entity. (Grierson, 2008)

team after the optimization process has ended. Pareto post-balancing methods can thus be regarded as being composed of a pre-process to locate the performance objectives and to run the multivariate optimization, then the later post-process applying a second (heuristic) optimization procedure. This later process is usually an evaluation process by a human (qualitative) interpreter, where every (pre-optimized) solution is graded. But most often, the post-process is simply seen as a “pick-and-choose” process of selecting the preferred solution among the many Pareto-ranked trade-offs, and this does not really account for an optimization.

We have now established the term optimization and what is required to perform an optimization. To understand how optimization will be combined into integrated dynamic models we must establish how optimization is actually used in practice.

3.2 Optimization in research and practice

Optimization of building performance is rarely used in the early design stage in practice (Kataras, 2010). And most research on applied optimization either occurs in the later stages specifically focusing on a small number of design variables having impact on building performance, or else optimization is used for a purpose unassociated with building performance, e.g. form generation. In addition to this, most of the research in the field of early design stage optimization restricts optimization to uncomplicated geometry which c*onsiders the building as a primitive or polygonal shape* (Kataras, 2010). Thus, it narrows the opportunity to establish a valuable connection between optimization and the early design stage.

3.2.1 Existing optimization tools and methods

In general, the methods that do apply optimization in early design stages focus on simple geometry or non-geometrical variables such as changing thermal transmittance or system requirements; they rarely put the analyses in context of project specific architectural needs. Many optimization tools, which are coupled to one or multiple BPS tools, have been developed during the past decade. Opt-E-Plus (Nrel, 2013), GENE_ARCH (Caldas, 2008), BEopt™ (Christensen et al., 2005) and TRNOPT (Bradley and Kummert, 2005) are examples of tools developed mainly for building energy performance optimization. These tools provide access to many different optimization algorithms; hence they serve as dedicated optimization tools for specific BPS environments. The advantage of this approach is a tight coupling of the two tools, which allows developers to create a more robust and feature rich coupled environment. Generic optimization tools such as Genopt (Wetter, 2011a) and MATLAB (Tonel, 2007) (normaly is MATLAB regarded as general mathematical modeling tool with many optimization options, but here referred to as tool which can be used to facilitate an optimization process) are developed to allow coupling to any computer software that is open enough to allow read/write interoperability. This approach has the advantage of flexibility, and the tools generally have a larger user base than the dedicated optimization tools.

The mentioned tool examples and other widely used tools for building energy performance optimization have been evaluated in a recent review by Palonen et al. (2013) Some of the key abilities of the tools were identified as follows:

- Support of multi-objective algorithms
- Automatic constraint handling
- Parallel computing
- Simultaneous handling of discrete and continuous design variables

In Figure 27 the key abilities of optimization tools are listed (the cost of the tool is not considered a feature, but may be an important factor for developing a larger user base). In addition to the above list the optimization tool MOBO (Palonen et al., 2013) should be mentioned, as it supports all the key abilities.

Customized and Generic Optimization Tools

	Optimization Tools	Q1	Q2	Q3	Q4	Q5
Customized	Opt-E-Plus	Yes	No	No	No	No
	GENE_ ARCH	Yes	Yes	No	No	No
	BEopt™	Yes	No	No	No	No
	TRNOPT	No	Yes	No	No	Yes
	MultiOpt2	No	Yes	Yes	?	Yes
	jEPlus+EA	No	Yes	No	Yes	No
Generic	GenOpt	Yes	No	No	Yes	Yes
	Model-Center	No	Yes	Yes/ No*	Yes	No
	modeFRONTIER	No	Yes	Yes	Yes	Yes
	DAKOTA	Yes	Yes	Yes	Yes	Yes
	iSIGHT	No	Yes	Yes	No	Yes
	MATLAB Optim-ization Toolboxes	No	Yes	Yes/ No*	Yes	No

Figure 27 Features of optimization tools, from (Palonen et al., 2013). Q1: Is it a freeware? Q2: Does it include multi-objective algorithms? Q3: Does it handle constraint functions automatically? Q4: Does it allow parallel computing? Q5: Can it handle discrete and continuous variables simultaneously?

Only GENE_ARCH is specifically developed for the early design process of building design. The tool is dedicated to the BPS tool DOE2.1E (James J. Hirsch & Associates, 2013) (on the side note this BPS tool can in many ways be compared to the Danish BPS tool Be10 (SBI, 2013)). GENE_ARCH is the only tool that has implemented dedicated rule based algorithms, also known as shape grammar (Stiny, 1980). This allows the user to apply advanced geometry-focused searches coupled to energy performance. The limitation of the GENE_ARCH tool is the high levels of coding experience required from the user. Every aspect of optimization, including geometry, needs to be encoded with Lisp.

Shape grammar is one of the primary features of integrated dynamic models (Negendahl, 2015a). Therefore, integrated dynamic models with optimization may be better compared to GENE_ARCH than any of the other tools. Today, integrated dynamic models with optimization are often associated with Rutten's optimization tool Galapagos (Rutten, 2010). Rutten is the main developer of the VPL Explicit History, now known as Grasshopper (Robert McNeel & Associates, 2013b), and Galapagos is completely integrated in the VPL. Galapagos is known for its implementation of evolutionary algorithms (see Section 3.3). Many building designers and researchers have used this implementation out of the box to optimize form, performance and other rule based searches through a parametric model. The benefit of using integrated dynamic models over dedicated tools for optimization is basically the same as why integrated dynamic models are more applicable to the early design stage compared to dedicated BPS tools. This was discussed in Section 2.5.

Integrated dynamic models have been coupled with generic optimization tools such as MATLAB for optimization (Negendahl et al., 2014; Trubiano et al., 2013). Therefore, any algorithm handled by MATLAB may be applied to integrated dynamic models. In addition to generic optimization tools coupled to integrated dynamic models, VPLs have in some cases their own dedicated optimization tools. These dedicated optimization tools are often called *plugins* that allow easy integration with existing VPL script environments. In Section 3.3 some of the most frequently used optimization tools and their algorithms coupled to integrated dynamic models are listed.

3.2.2 Criteria for the optimization approach

Obviously, compulsory and ambitious use of optimization algorithms in the early design stage is of architectural concern. Hermund (2009) reacts towards optimization in the design processes:

> "*Linear working methods that promote the reduction of the creative loops in favor of systemic optimization is one topic that must be addressed by architects … Relying on one integrated model (referring to IFC- and gbXML-models) could mean an eventual loss of control with real value of the architectural quality: to create meaningful and beautiful spaces for real people.*" Hermund (2009)

The concern of using optimization processes in early design is very real, regardless of how a model is constructed, or of the number of design variables and number of quantifiable objectives. However, the benefit of optimization may in many cases exceed the downsides of artistic control, if the optimization processes are controlled and supervised by the designers themselves (Caldas and Norford, 2001). Therefore, the criteria for optimization are first and foremost connected to the goal of better supporting the early design stage. This includes the ability to optimize geometrical design concepts representing architectural ideas. Based on Mora et al. (2008) and Struck et al. (2009) optimization is best supported when the approach is able to:

- Assist rather than automate design
- Facilitate the quick generation of integrated solutions
- Shorten synthesis analysis and evaluation cycles
- Support an interaction and selection of most suitable design alternatives

If optimization is to be used in the early design stage, it will be used to *assist* the design team. Therefore, integrated dynamic models with optimization will only be *part of* the design process and not *the* design process. Also the definition of *integrated solutions* is open for discussion. Are integrated solutions purely based on performance criteria? Or is the inclusion of qualitative objectives part of the *generative process*? These questions will be discussed in Section 3.4. But before this is clarified, a brief summary of the currently available optimization algorithms to support Consequence based design is given.

3.3 Optimization algorithms

There are a wide range of optimization algorithms available for building designers and simulationists today. In the following, some of the most frequently used algorithms are briefly reviewed. All algorithms are available for integrated dynamic models, either through dedicated optimization tools (plugins) for the VPL for example: Galapagos (Rutten, 2010), Octopus (Vierlinger, 2014) and Goat (Simon Flöry et al., 2015), or through generic optimization tools such as MATLAB (Tonel, 2007).

In general terms there are two types of optimization algorithms; deterministic and stochastic algorithms. The deterministic algorithms such as brute force search methods are much slower than the stochastic methods. Stochastic search methods rely on random elements to generate unique outcomes of complex, multi-parameter problems. The randomness may result in unique outcomes with no guarantee of finding the exact optimum. Non-stochastic methods, on the other hand, are entirely deterministic in their nature. They are generally more reliable for finding the precise optimum since they do not get stuck within local minima or maxima. There is no degree of *creativity or serendipity in the outcome of deterministic optimization* (Wilkinson, 2011). Wilkinson (2011) even argues that stochastic search methods are more applicable to the design world, whereas deterministic optimization methods suit the engineering side. In the combined effort of a design team, this author argues that both deterministic and stochastic optimization have their justification in the early design stage.

Deterministic search methods

Deterministic search methods include heuristic search, complete enumeration, and random search techniques. Heuristic optimization is often associated with manual optimization where the user changes parameter settings and design variables and then makes a simulation. This process continues until the analyst believes that the output has been optimized. One example of such a method is described and discussed by Petersen 2011 using the BPS iDbuild (Petersen and Hviid, 2012). Random search techniques often utilize uniform or normal distributions that center a symmetric probabilistic density function. Deterministic search methods are also sometimes known as Brute force techniques.

Direct Pattern search methods

Direct Pattern search methods neither compute nor explicitly approximate derivatives of cost functions. Thus, the unifying theme that distinguishes pattern search from other (direct) methods is that each of them performs a search using a “pattern” of points independent of the cost functions (Torczon, 1997). The best known Direct Pattern search algorithm is Hooke-Jeeves (Hooke and Jeeves, 1961). Variations of this algorithm are found in many implementations such as MOBO, MATLAB, and GenOpt. Under the assumption that the cost function is continuously differentiable, all accumulation points constructed by the Generalized Pattern Search algorithms are stationary (Wetter, 2011a).

Newtonian search methods

The (Damped) Newton approaches are often used in discretization of large systems of nonlinear algebraic equations (Dirkse and Ferris, 1996). In the cases where the convergence of the Newton scheme is attainable only for very small time steps, methods for the enforcing of the convergence are often applied. An example of a Damped Newton search method in combination with optimization of building design performance is showcased by (Pedersen, 2006), who also concluded that the inclusion of damping terms can reduce the number of iterations significantly.

Evolutionary search methods

Evolutionary search methods, also known as Genetic Algorithms (GAs) and Simulated Annealing algorithms (SAs), are some of the widely used optimization methods in building design. The algorithms are versatile but can be difficult to predict because of the many hyper parameters that control them. SA borrows its basic ideas from statistical mechanics: A metal cools, and the atoms (design variable vectors) align themselves in an “optimal state” for the transfer of energy. In general, a slowly cooling system, left to itself as it eventually finds the arrangement of atoms, which has the lowest level of energy state. The “cooling” behavior is what motivates the SA, as it converges towards an optimum. GAs are probabilistic optimizing algorithms that like SAs do not require mathematical knowledge of the response surface of the system. They borrow the paradigms of genetic evolution in nature, and utilize the hyper parameters: *selection, crossover*, and *mutation*.

- **Selection:** The current solutions are defined by points in hyperspace and ranked in terms of their fitness by their respective response values. A probability is assigned to each point proportional to its fitness, which determines a portability to mate the most promising solutions in pairwise configurations (selection of the fittest parents).
- **Crossover:** The new point, or offspring, is chosen, based on various combinations of the genetics (combinations of design variable vectors) of the two parents.
- **Mutation:** The offspring is also susceptible to mutation, a process that occurs with probability p. In this case, the offspring is replaced randomly by new combinations of variable vectors.

Agent based search methods

Agent-based models (ABMs) for optimization and Particle Swarm Algorithms (PSAs) are not as commonly used in building design as e.g. genetic algorithms. However, agent based search methods have gained increased interest in other design areas such as in transportation and manufacturing industries (Barbati et al., 2012). ABMs and PSAs are often said to be inspired by the social behavior of organisms such as fish schooling and bird flocking. Kennedy and Eberhart (1995) explains the agents as *particles assigned with flocking behavior in hyperspace to look for the optimal position to settle*. Each individual, namely particle, is assigned with a randomized velocity flown through hyperspace. PSAs are almost identical with ABMs. However, PSA agents are individual solutions roaming in a competitive space of solutions, whereas ABM agents are usually competing in same-state solution space.
Nowadays, PSAs and ABMs have gained much attention and wide applications in solving continuous non-linear optimization problems (Eberhart and Yuhui, 2001). However, the performance of PSAs greatly depends on their hyper parameters, and similar to GA and SA, they often suffers from being trapped in local optimum (Liu et al., 2005).

3.4 Integrated dynamic models and optimization

Integrated dynamic models with optimization are basically an extension of the integrated dynamic model with a coupled optimization tool as shown in Figure 28. The optimization tool in most cases (one exception is Moth, see Section 4.4.5) refers to the integrated dynamic model by duplicating instances of the data model, thus it receive a static representation of the model along with how the model performs at this particular instance. This representation is composed of a particular combination of design variable vectors. The optimization tool also needs defined objectives. Often, all objectives must either be minimized or maximized. Constraints may be defined in any of the coupled tools; often this is done by constraining the design variables in finite ranges in the VPL. The performance feedback from the BPS tool may be redirected directly to the optimization tool, or it may be subject to any constraints, weights or rule-based modifications performed in the VPL. In this way, the VPL may be used to aggregate multiple objectives from multiple BPS tools. Or it may be used to create complex *synthetic* evaluations such as utilizing shape grammar in the constraint functions.

The inclusion of optimization in integrated dynamic models has already shown great promise. For example, Sheikh & Gerber (2011) showed that it was fairly easy for a building designer to implement constraint functions and couple this to Galapagos and a BPS (DIVA (Jakubiec and Reinhart, 2011)). And more recently Lauridsen & Petersen (2014) has used Galapagos with multiple BPS tools (DIVA (Jakubiec and Reinhart, 2011), Viper (Energy+ (U.S. Department of Energy, 2013)) and ICEbear (BuildingCalc (Nielsen, 2005)). There is little doubt that the VPL can facilitate optimization algorithms, and therefore, combining integrated dynamic models with search methods is fairly straight forward. However, just because optimization tools such as Galapagos are available and uncomplicated to implement, to apply optimization in integrated dynamic models in practice is not uncomplicated.

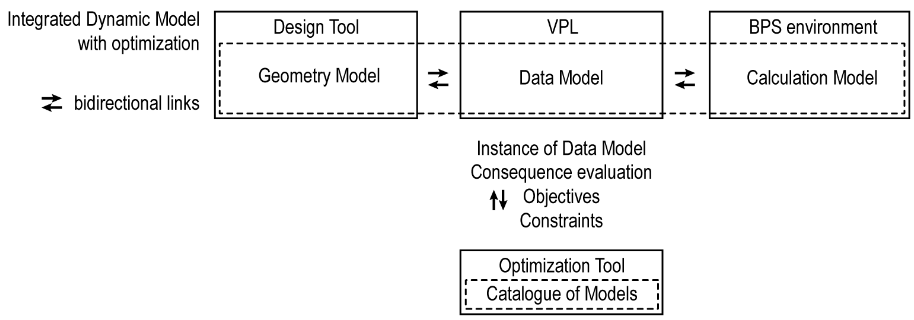


Figure 28 Integrated dynamic models with optimization consist of a design tool, a VPL, one or multiple BPS tools and an optimization tool.

With optimization new challenges arise. Hermund (2009a) is one among many building designers who raise alarm about allowing machines to take over the design process. This, however, is a general concern and is not exclusively related to integrated dynamic models with optimization. When optimization is applied to integrated dynamic models, what remains even more important is quality assurance. As optimization algorithms use BPS output as input for the cost functions, the search is very sensitive to errors in the BPS output. Algorithms, at least the ones we have today, have no understanding of what an objective is and it does not know the concept of analytical and critical thinking - also known as "common sense". Thus, an algorithm cannot distinguish *bad* input from *good* input, and it will not hesitate to present optimal solutions that are far from actual optima if the model is poorly set up. The whole process of setting up a model to be able to handle any type of error, and of quality assessing every thinkable combination of design variable vectors, is much more time consuming compared to integrated dynamic models without optimization. Of course, this amount of time could be matched when a machine searches for answers. Machines, even with computational inefficient optimization algorithms, are considerably faster than humans in this regard. But again, one minor mistake or misconception in the problem definition will propagate into the cost function and deliver biased and sometimes completely wrong optima. The problem is to define the problem.
Hermund (2009a) who among others cares about qualitative objectives such as aesthetics, functionality, etc. will probably raise similar concerns about utilizing any type of automation to represent qualitative objectives. It is likely that any type of machine automation will lead to *"an eventual loss of control with real value of the architectural quality".* Nevertheless, the presence of the VPL in the integrated dynamic model allows objectives such as aesthetics to be incorporated *"as soft elements in any decision support system",* as argued by Harding et al. (2012). However, doing this in practice is not trivial. Trubiano et al. (2013) is one of the

most recent examples of research in architecture, High-Performance Buildings and the use of integrated dynamic models in combination with optimization algorithms. The authors used the VPL Grasshopper linked to OpenStudio (NREL U.S. Department of Energy, 2014) (Energy+ (U.S. Department of Energy, 2013)) and Radiance (Perez et al., 1990); an evolutionary algorithm was coupled through MATLAB (Tonel, 2007) to maximize solar radiation levels in the heating season and minimize solar radiation in the cooling period; a penalty function was used to control illumination levels. Even though the authors argued that by defining the design parameters subjectively they managed to integrate the analysis of qualitative objectives*,* they acknowledged that there are still many challenges in combining the objective and subjective decisions in optimization. These challenges can be divided into two categories:

- Methodical optimization challenges
- Optimization and BPS tool challenges

3.4.1 Methodical optimization challenges

Few researchers address the complicated methodical challenges in the need to combine human evaluation with computational speeds in early stage optimization, which include the way machines may interact with human qualitative objectives and vice versa.

> *"…designs do not result from performance requirements in the way that effects are thought to result from causes."* (Mahdavi, 2004)

A popular method to apply optimization in design is based on parametric models in combination with stochastic optimization algorithms. Often, these algorithms are utilized with the purpose of *generating* solutions based on the designer's own objectives, which may or may not be coupled to building performance related attributes. Sometimes, multi-objective fitness functions are used for *something else* than pure optimization, and they are applied to ill-defined[16] problems such as art, music and computer graphics. This makes optimization act as an exploratory tool as opposed to its primary function as an optimizer tool (Byrne et al., 2011b). In these instances, form generation and aesthetical evaluations are the main performance indicators and often solutions are *molded* into the designer's liking.

Artistic control

To maintain an *artistic control* of the optimization algorithms, many different attempts have been made. One dominating idea is to let the designer control the objectives rather than the actual geometry. Researchers such as (Gaspar-Cunha

[16] Several alternative responses might be made to the issue of ill-definedness. At one extreme, one might conclude that the quest for systematic automated design procedures in building design is fruitless in principle. At the other extreme, it might be argued that the appearance of ill-definedness is simply the result of sloppy thinking and inadequate information, and that this may in principle be remedied as more adequate problem formulations are evolved. (Mitchell, 1975). The author's stance is that ill-defined problems have an undetermined dimension, and human decisioning is necessary to make decisions and thus to solve ill-defined problems.

et al. 2011) and (Caldas & Norford 2001) argue that weights on objectives give the designer the highest level of control over the optimization process. However, this does not change the fact that the designer loses control to an algorithm that is solely focusing on predefined objectives. And obviously, changing performance objectives to generate an outcome that is more pleasing to the designer does not make sense, if the main criterion is to maximize performance of the predefined objectives in the first place. Caldas & Norford (2001) also notices that applying weighting factors to attributes (design variables), will to some extent constrain the design process itself. Partially because changes of weighting factors may be discontinuous, thus leading to dramatically different solutions. And partially because setting up weighting factors requires a large amount of insight into all factors involved in the decision-making process. This is coupled to the constraint handling associated with Penalty functions and Barrier functions as discussed in Section 3.1.1 and the quality assurances discussed by Hensen (2002) and Augenbroe (2002).

Locked-in explicit history

The challenge in using optimization algorithms in the early design stages is in many ways the challenge to maintain the support of the feedback process during the optimization cycles.

> *"Evolutionary solvers tagged onto parametric models (in order to enable cyclic graphs) do not address this inflexibility at the meta-level, that is, the topology of the symbolic graphs itself, instead reinforcing the belief that because something has been 'optimised', it must be a good design solution – this is a fallacy."* (Harding et al., 2012)

Some of the critique on using parametric tools for optimization is found in the limitations of the tools used today. Harding et al. (Harding et al., 2012) argues that parametric models at present are no different to CAD- or indeed BIM-models, as designers usually only deal with singular building typology. They state that the current VPLs like Grasshopper and Dynamo basically just allow the designer to assemble typology in a *symbolic graph structure,* instead of fixed geometry. This graph structure is viewed top-down and has to be comprehensible by a single human brain in order to be tweaked and modified. This is done in a linear fashion. Harding et al. (2012) proposes the use of *complex adaptive systems* (generative modeling techniques) without a strict discrete structure. Apparently, this will allow bottom-up rules giving rise to emergent structures with no locked-in *explicit history* (Harding et al., 2012). This type of parametric thinking is still within the boundaries of an integrated dynamic model, and there is no real difference in coupling BPS tools to an adaptive parametric model (associated with the generative model illustrated in Figure 11). However, at the present stage, the VPLs most widely used (Grasshopper and Dynamo) are based on top-down techniques. Nevertheless, the use of VPLs and integrated dynamic models is still much better at representing the needs of the building designer, than any dedicated optimization tool existing today.

The rigid optimization algorithms

Design may be subject to optimization with traditional stochastic algorithms (e.g. SAs and GAs) as long as design constraints and objectives are both constant and lie within a measureable metric. In reality, this assumption is incorrect *because initial constraints and objectives rarely stay constant* (Harding et al., 2012) in the early design stage. The ParaGen (Buelow et al., 2010; Turrin et al., 2012) project is a good example of this, relying on the use of GAs not only for optimizing geometry, but also for more extensively exploring the solution space. In this respect, the process benefits also from information extracted from sub-optima as well as poorly performing solutions. If the chosen solution in the early design stage is actually sub-optimal, this might later cause *cost overruns and the requirement to do additional work – something the parametric model was initially meant to prevent* (Harding et al., 2012). When optimization is introduced in the integrated dynamic model, the flexibility of the model is thus even more important. Harding et al. (2012) points towards an example from practice where the design team had altered the conceptual design to a point where *the parametric model schema could not cope with the changes.* Often, the VPL itself was determining an inflexible method of working which is incapable of responding to changes outside of its system. Instead of going back to *square one*, Harding et al. found it necessary to *hack away inefficiently at the parametric topology* until the model acted as it was supposed to. Such last resort is therefore the only option available due to time constraints. When optimization algorithms are used in the early design stage, the time constraints may be even more limited and therefore in practice optimization is not completely uncomplicated as it might be in research projects.

GAs, SAs, PSAs and most other stochastic optimization algorithms applied to integrated dynamic models are during optimization roaming in a competitive space of solutions also known as the hyperspace. This means that when the algorithm evaluates which solution is better, it needs to compare more than one solution. And thus it handles solutions as populations or groups in the hyperspace. The challenge is if the designer wishes to alter, for instance, the weighting factors to guide the design towards a qualitative objective such as aesthetics, the designer needs access to the algorithm during optimization. It is very difficult (mathematically) to apply subjective changes to hyperspace populations and it is slow to apply the user alterations to the currently “best solution of the population” for every iteration. Therefore, significant changes to the way optimization algorithms work may be needed to better support user interaction during optimization.

One approach could be the utilization of in-state optimization processes as the recently implemented ABM method showcased by Negendahl et al. (2015). However, as noted by the authors, at the present state, more robust implementations and further research of such methods are needed if qualitative objectives are to be handled by in-state during optimization algorithms.

3.4.2 Optimization and BPS tool challenges

To complicate matters further the BPS tools used to feed the optimization algorithms affect the actual outcome of the optimization process. Indeed, the type, precision and quality of the BPS tool matter to a great extent. But what is less obvious is that most BPS tool solvers approximate solutions due to adaptive variations in solver iterations (Wetter and Polak, 2004). Thus, they tend to form dis-continuous search spaces. As Wetter (Wetter, 2011b) explains non-linear programming algorithms (implemented in the iterative solvers in most BPS tools) are computationally efficient. However, these solvers need to compute a numerical approximation $f^*(\varepsilon, x)$ to the cost function (of the solver), where ε is the tolerance of the numerical solvers. The main challenge with optimization when combined with BPS tools according to Wetter is the discontinuities in the numerical approximation to the cost function. This is due to the fact that a change in design parameters can cause a change in the number of solver iterations. Tightening the solver tolerance ε is possible in most BPS tools, but this may increase the computational load significantly, and also this is usually not allowed from the BPS tool.

To balance speed and precision may be the most difficult choice for the designer when utilizing early design stage optimization. On the one hand, the fast feedback speed provided by simplified BPS tools may be required to perform enough solutions within the limited timeframe of the early design stages. On the other hand, the more advanced BPS functionalities are slower in comparison to the simpler methods. But this in turn generates more precise feedback, which may be required to satisfy the optimization algorithms available to the designer. Few BPS tools are built to support early design stage energy optimization and unfortunately, many optimization algorithms have difficulties in handling the BPS discontinuities (Wetter and Polak, 2004). Therefore, it is not a simple task to couple a BPS tool to an optimization tool, which of course applies to integrated dynamic models with optimization. According to Wetter (2011b), the best one can do in trying to solve optimization problems, where the cost and constraint functions are evaluated by a BPS, is to run multiple instances of the optimization problem. Numerical experiments (Wetter, 2011a) show that by using tight enough precision and by starting the optimization algorithm with coarse initial values, one often comes close to a minimizer of $f(\cdot)$. Furthermore, Wetter suggests selecting different initial iterates for the optimization, or by using different optimization algorithms, increasing the chance of finding a point that is close to a minimizer of $f(\cdot)$. Nonetheless, even if the optimization terminates at a point that is non-optimal for $f(\cdot)$, *one may have obtained a better performance* (of the relevant objectives) *compared to not doing any optimization to begin with* (Wetter, 2011a).

The computation time required by an annual building simulation depends on many factors as discussed in 2.4.2, and can last from less than a second to several hours. A typical optimization analysis requires a few hundred to several thousand evaluations until a good solution is reached, depending on the selected algorithm and the size of the solution space. This results *in processing times varying from some minutes to hours or even days* (Machairas et al., 2014).

Selecting the appropriate approach and applying specific techniques can significantly decrease the simulation runtime. Machairas et al. (2014) attempted to find the best performing optimization algorithm for building design, but could not find a general conclusion: *As the stochastic behaviour of the algorithms, and the plethora of configuration options for each algorithm, as well as the diversity of building design problems, no safe conclusions about the performance of optimization algorithms can be found*. Nonetheless they found that Direct Pattern search methods, GAs, SAs and PSAs have been successfully coupled BPS tools. Direct Pattern search can be very efficient if the objective function doesn't have large discontinuities; *otherwise it can fail or got trapped in local minima*. A good approach is to use an evolutionary algorithm for global search and a Direct Pattern search to refine the solution (Machairas et al., 2014).

3.5 Conclusion

The vast majority of problems within the early stages of building design involve balancing many quantitative and qualitative objectives. Most research that seeks to merge the optimization of building design with building performance, utilizes techniques that involve pre- and post-processing of the optimization process. This includes the pre-process of clearly defining objectives and boundaries, and a post-process of selecting the *most suitable design alternatives* (Mora et al., 2008). Alternatively (or in combination with), by adding and controlling weights on cost functions (as preferred by Gaspar-Cunha et al. 2011; Caldas & Norford 2001), the design team may change the direction of design (generation) toward qualitative objectives such as finding more aesthetically pleasing solutions in the search space. However, putting weights on objectives is a very complicated task and it may contradict the purpose of optimization.
To counter the discerning nature of singular weighted optimization methods, different multi criteria optimization approaches have been explored. One approach in particular that has gained much interest in recent years is the use of non-dominated optimization methods, often referred to as Pareto optimality (Grierson, 2008). Pareto methods help to identify a set of feasible designs that are equal-rank optimal. In this process, all objectives are defined as competing, un-evenly balanced criteria. Pareto methods are either defined by post grading the objectives, e.g. (Liu et al., 2012), or by seeking an equally good solution (in terms of non-dominancy) as suggested by different authors (Byrne et al., 2011a; Ellis et al., 2006; Fialho et al., 2011; Gerber and Lin, 2012; Limbourg and Kochs, 2008; Machwe and Parmee, 2007; Mela et al., 2012; Shi, 2011; Wang and Zmeureanu, 2005). Most approaches using Pareto optimality utilizes various types of constraints or other methods to balance objectives by posterior means. In general, all approaches share a common critique: they all rely on clearly defined criteria and boundaries, all of which are directly translated to objectives and constraints.

To apply optimization to building design is challenged by the way BPS tools are constructed. It is challenged by the way design frequently changes the direction of design. It is challenged by the available optimization algorithms and it is challenged by the limited time in the early design stage. Nonetheless, Integrated dynamic models is a small but steady a move away from the early rather rigid instances of optimization applications and code intensive generative systems that have dominated the optimization approaches just a few years ago. Integrated dynamic models add a more responsive stance to the specific traits of the design process, and according to several researchers (Buelow et al., 2010; Lauridsen and Petersen, 2014; Sheikh and Gerber, 2011; Turrin et al., 2012) the VPL assures more qualitative objectives to be accounted for. However, such stance does not indicate a fundamental change in the basic computational techniques underlying optimization approaches (Mahdavi, 2004), but rather an intention to incorporate such techniques in the overall context of a design support environment (Mahdavi et al., 1997). Utilizing integrated dynamic models makes it easy for the designer to automate processes that include High-Performance Building optimization. By contrast to dedicated optimization tools the integrated dynamic model thus allows the designer to optimize on creatively constrained design problems; it also gives the option to implement new automation algorithms such as agent-based models (Negendahl et al., 2015). Therefore, it can be concluded that optimization with little effort can be included in integrated dynamic models. Thus, integrated dynamic models with optimization can be used to find optimal High-Performance Buildings. But in terms of handling both the qualitative and the quantitative objectives, unbiased computer automated building performance optimization is difficult to put into practice.

4. Consequence based design

The first chapter outlined the potentials and the background of Consequence based design, the structure of tools and the structure of the integrated dynamic model itself. Chapter 3 introduced optimization as a concept and how this relates to Consequence based design. This chapter demonstrates and discusses the findings of Consequence based design when applied in practice.

During the three year project the author has been part of several building design projects representing Grontmij as an active participant in the design teams. The author has been either in charge of or partly responsible for the analysis and evaluation[17] of the building designs concerning performance related requirements such as energy consumption and/or various parts of the indoor environment. Some of these projects have been subject for the Consequence based design approach, five of which are presented, evaluated and discussed in this chapter. In addition to these five presented case studies the author has been either the singular developer or part of a team in the development of new tools, methods and algorithms in the attempt to support the Consequence based design approach.

Firstly, five selected case studies of Consequence based design are demonstrated. These case studies have been real projects involving different stakeholders and disciplines. Secondly, practitioners' perspectives are discussed along with the findings from the five case studies. Then selected developments (algorithms, tools and methods to support the approach of Consequence based design) are presented. These Developments are various attempts to support and improve integrated dynamic models and the approach of Consequence based design.

[17] Following Souza's (de Souza, 2009) terminology: the author acted as a *simulationist* who process and analyze data from simulation and calculation tools.

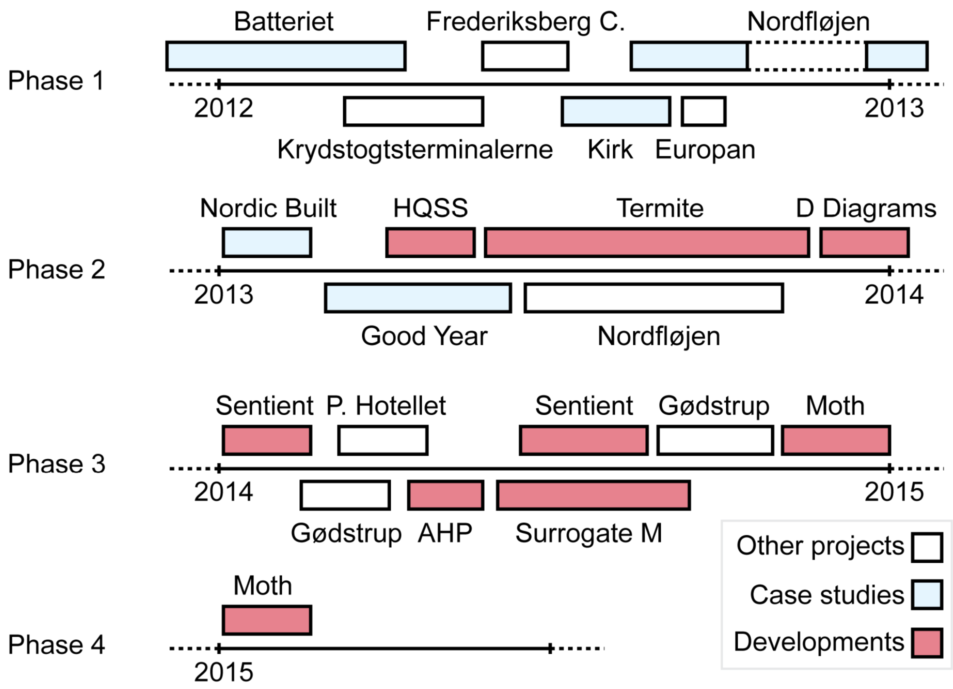


Figure 29 An overview of the projects that have been involved in forming this thesis

The project is divided into four phases, where Phase 1 focused on case studies, see Figure 29. Phase 2 was a transition phase from project specific use of integrated dynamic models into developments of universal applications of Consequence based design. Phase 3 was dedicated to further development and documentation of the universal applications of Consequence based design. And finally Phase 4 was mostly dedicated to document the findings and write this thesis.

Three types of projects are identified that have formed this thesis. In blue (Figure 29), the timeline is visualized of selected case studies, where the Consequence based design as an approach has been used by Grontmij and their partners. In white, the timeline is visualized of projects that have influenced this Ph.D.-project, but where documentations or detailed results have been insufficient to present in the thesis. In red, the timeline of Developments are shown. These are developments that are not directly associated with a specific commercial project, but could be universally applied to other projects. These developments list ontology algorithms, software tools, prediction algorithms and optimization algorithms and various methodical approaches to qualify early design decisioning.

4.1 Case studies - the application of integrated dynamic models

The five case studies chosen to be presented in this thesis all share the use of one or multiple integrated dynamic models. The chosen case studies are used to test and document different aspects of the integrated dynamic models. For example case study Kirk is used to test the geometric model complexity of an integrated dynamic model and the case study Good Year was used to test the application of multivariate optimization during the early design stage. In this way various concerns and challenges of using integrated dynamic models and Consequence based design are exposed and discussed. In Table 3 selected focuses on each integrated dynamic model are summarized. The five cases cover subjects such as *collaboration, validity, decision making, optimization* and more. None of the cases were specifically designed to accommodate the analysis of a specific focus of an integrated dynamic model. The analyses were on the other hand retrofitted during or after completion of the case study. That said as the author has been part of the design process in some of these projects the inclusion and structure of the utilized integrated dynamic models have been unavoidably affected by the author. The author's role in each case study has been stated in the beginning of each case description.

Table 3 Selected focus on the integrated dynamic model in each case study

Case study	Selected focus of the integrated dynamic model
Batteriet	Collaboration, responsiveness, economic benefits
Kirk	Model complexity, responsiveness, validity
Nordfløjen	Collaboration, Multiple BPS tools, analysis complexity
Nordic Built	Quality defined objectives, collaboration, decision making
Good Year	Optimization, multiple objectives, analysis complexity

In an attempt to collect the findings of the five cases concluding remarks on the case studies is found in section 0. Here the integrated dynamic models used in the case studies is discussed (when relevant) in terms of:

1. Flexibility (of features and complexity)
2. Speed (of performance feedback)
3. Precision (of the BPS)
4. Usability (i.e. easy to use)
5. Visual quality (for presentation)

Second, the use of the Consequence based design approach is discussed in terms of how the design teams used the integrated dynamic model. The focus is centered on the integrated dynamic models used as a:

6. Sketching tool
7. Communication tool
8. Calculation tool
9. Collaboration tool

These projects were carried out with a network of different stakeholders, and not all stakeholders have been interested in being published. To respect this, and to remain focused on the *development* and *innovation* of the research, some of the projects or parts of the projects have been anonymized.

In the following sections, the commercial value and the research value the projects must be clearly distinguished. The documentations presented here are primarily focused on the research values, and in this spirit the specific projects must therefore be seen as means to emphasize the findings relating to the Consequence based design approach.

In the aim to collect and compare information on Consequence based design and the application of integrated dynamic models, structured surveys have been conducted on the five projects. One example is found in Appendix B. These surveys are conducted along with interviews and collection of internal documentations (primarily from Grontmij) such as strategy assessments, emails and project evaluation reports. Every member of each design team participating in the case study projects has been invited to participate in the specific surveys. However, not every survey has been equally successful in collecting adequate data from the team. Therefore, the results of the surveys must be read in the context of the other findings.

The interviews have all been informal and based on discussions on the individual projects.

In the surveys, the following three questions were given to the members of the design team:

- *“Have the interactions between members of the design team been affected by the use of integrated dynamic models in this project? If so, how?”*
- *“Has the use of integrated dynamic models had an impact on the simulated performance such as energy consumption, indoor environment and/or sustainability of this project? If so, how?”*
- *“Has the use of integrated dynamic models had an impact on other decisions in this project? If so, which?”*

None of these questions mentions Consequence based design as an approach. This was not found necessary as the term is invented by the author. The questions, however, revolve around the use of integrated dynamic models in ways that can help the approach of Consequence based design. Each answer is found in the Appendix. The most impactful answers are presented directly or written as parts of the results for each case study.

The members of the design team were also asked to answer to what extent they agree on the following five statements A), B), C), D) and E) on a scale from 0 to 100. Where 0 is “I do not agree” and 100 is “I agree completely”. The five statements are:

- **Statement A)** The use of integrated dynamic models has created a better common starting point of collaboration between the members in the design team

- **Statement B)** The use of integrated dynamic models has improved the communication between the members in the design team

- **Statement C)** The use of integrated dynamic models has improved the (simulated) performance of the building design
- **Statement D)** The use of integrated dynamic models has assisted in the exploration of architectural expressions and concepts of the building design that is not directly associated with the (simulated) building performance

- **Statement E)** The use of integrated dynamic models has positively assisted in achieving high performing buildings without compromising the architectural expressions and concepts of the building design

In the results for each case study the responses given to these statements are presented. The statements refer directly to the research questions and the aims and objectives as seen in Section 1.4 and Section 1.5. As mentioned in the Research method (section 1.8) these surveys and interviews do not provide statistical ground for general conclusions for the entire industry. However they may point towards trends and individual findings may provide value for the general knowledge of integrated dynamic models.

All case studies have been carried out with Grontmij as an engineering consultant with different architectural studios. The projects range from sustainable social housing renovations to large scale hospital projects. In the following sections, each project is briefly described, then the project is discussed in the context of Consequence based design, and finally each section concludes on the main findings of the project.

4.1.1 Case study: Batteriet

Batteriet, "the Battery" in English, is a mixed (office/housing/commercial) project developed by BIG and various consultants. This is not only the first project of this Ph.D. study but the first real project where Grontmij attempted to utilize an integrated dynamic model in a Consequence based design process. Therefore, many of the concepts of Consequence based design were in early development during this project and the definitions and terminologies were yet loose and unclear. Also this project was the only project where the author was not in charge of the integrated dynamic model. Of this reason many of the references and conclusions made by Grontmij has been established before and in parallel to this study. What is of special interest was the collaboration between members in the design team, and the very simple quasi-steady-state method developed for the particular integrated dynamic model. Also, many of the results from the project are related to the motivations defined in Section 2.1.3 and 2.1.4. The documentation is based on various internal and external reports, interviews and a survey. This case study parts itself from the other case studies by having a thorough set of internal evaluation documents and an internal business potential report build around the project. Therefore the focus of the case study is to demonstrate and discuss various organizational and design process traits and limitations related to the use of integrated dynamic models and Consequence based design. These documents are not part of the thesis due to confidentiality concerns.

The exact calculation method used in the integrated dynamic model is neither presented in detail nor discussed here, as some of the data was ineligible for publication for commercial reasons.

Introduction

The Battery is a building complex consisting of several large mountain-shaped buildings situated in Copenhagen, Denmark. In this case study, one of the buildings is analyzed and optimized according to an overall building heat-balance-equation developed by Grontmij. The project was unfortunately put on hold due to lack of funds with the client.

The whole approach into developing an integrated dynamic model was due to the very specific request of the building designers. They specifically asked for a parametric (Grasshopper) model, *which could deliver fast and flexible consequence feedback in the design tool*. The building designers were already utilizing parametric modeling in many of their projects. The building designers regarded the use of VPLs as a *natural part of the design process*. The request to the simulationists came out of the need of a concrete method to find a balanced window-wall-ratio (WWR) on the staircase-like sloping façade on the particular building, see Figure 30. The *unusual distribution of volume* and the sloping façades of the building made their current rule-of-thumbs to determine minimum and maximum allowed[18] WWR inaccurate and insufficient. The integrated dynamic model was to be created as a tool by the simulationists and handed over to the building designers for experimentation on various façade design strategies.

Figure 30 The size of each window is defined by a heat balance equation. The color determines if the winter situation (blue) or the summer situation (red) is more dominating. Green windows are balanced out.

A simple integrated dynamic model

The size, the unusual distribution of volume and the building designers' wish for a highly transparent façade of the case-study building were of great concern for the heat balance of the building. The simulationists argued that high risk of overheating and high levels of heat loss through the building envelope were pushing the limits of what was allowed in terms of the Danish building regulations (Danish Building Regulations, 2013). As the usual rules-of-thumb would not apply to this situation and the modeling required in BPS tools would be extensive[19], it was decided to model a form of heat-balance calculation (quasi-steady-state simulation, see Figure 31) for the entire building to analyze and control the amount of transmission through the transparent façade.

---

[18] The request from the building designers for minimum and maximum WWRs was mainly to avoid violation of the Danish building regulations (Danish Building Regulations, 2013)

[19] At this present time there were no existing couplings of BPS tools to VPLs, this has later changed significantly.

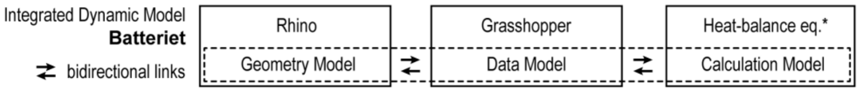


Figure 31 The integrated dynamic model. The design tool Rhino combined with the VPL Grasshopper and a heat-balance equation. *The heat-balance equation was developed and implemented in Grasshopper, and this acted as a BPS tool.

The heat-balance calculation was based on the assumption that minimal heat loss and a known average cooling capacity were the defining factors of the energy balance of the envelope. And if the heat-balance of each window could be determined, the minimum and maximum allowed size of each window could be found. The input relied on individual window areas, window orientation and window placement according to site context (Figure 32 and Figure 33). And the geometry model was based on the building designers' sketches and model specifications.

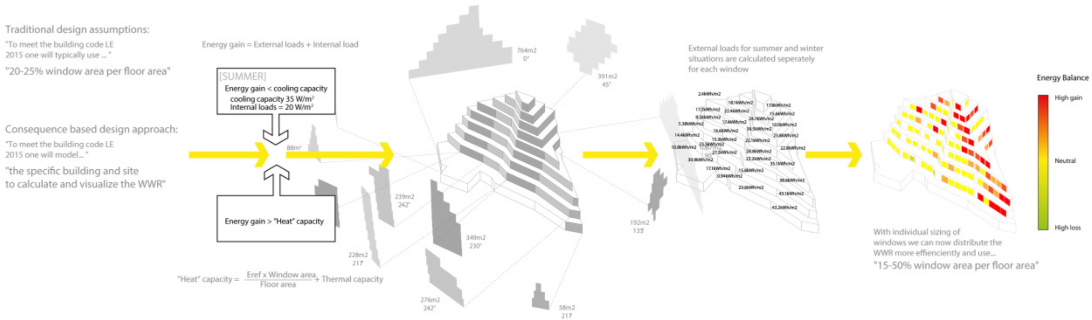


Figure 32 The Consequence based design approach related to the model as it was presented to the building designers in the design team. Image by courtesy of Grontmij©

The model was constructed with two graph-based user interfaces (using the VPL Grasshopper). One interface was dedicated to the building designers and one to the simulationists. In general, the aim was to maintain a high quality assurance by separating the heat-balance equation (variables), and to provide the building designer as much control as possible with the geometry. The building designers had parametric control of windows, which included a sorting algorithm that grouped similar windows based on their mean irradiance, E_w, their orientation and mean heat accumulation Q/A for each floor. The simulationists could control a weighted average of the *ideal sized* windows in the summer period over the ideal winter period (there quasi-steady-state method was based on two weather data sources). The properties such as g-value and U-values and specific variables concerning the heat-balance equation were reserved for the simulationists.

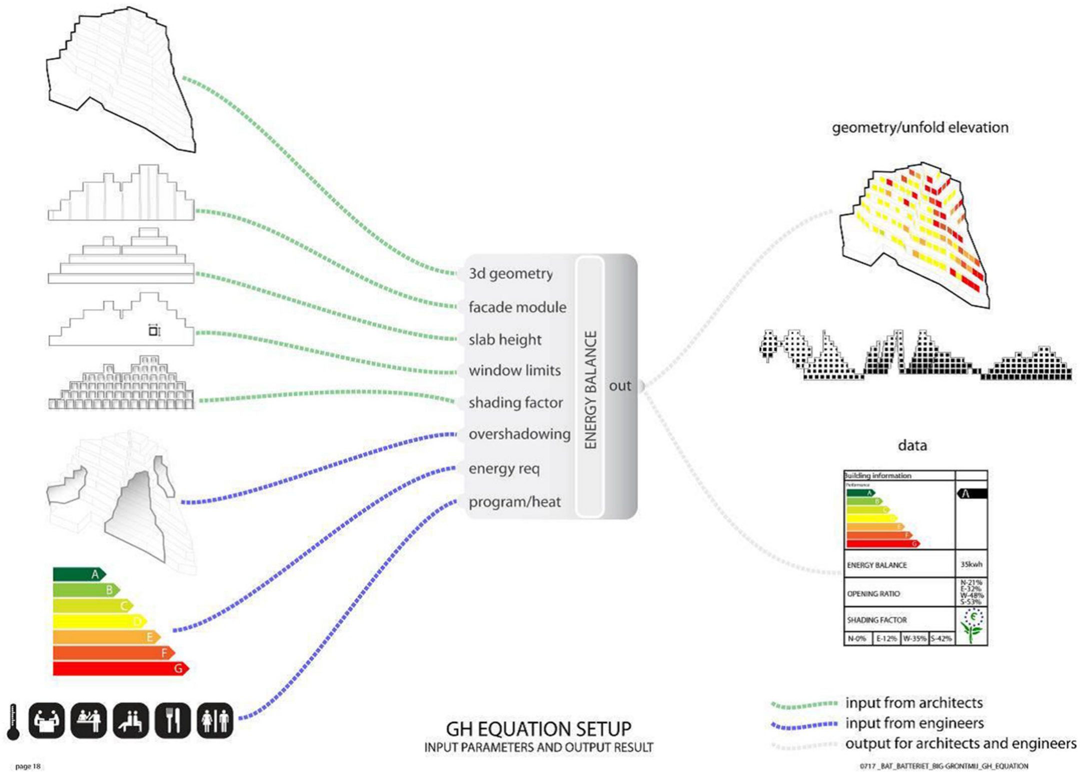


Figure 33 The integrated dynamic model acted as a black box tool with custom input (to the left) and custom output (to the right). Image by courtesy of Grontmij©

The visual outputs in the 3D model were generated by the model as squared windows colored by their weight between their ability to gain more energy or loose energy over the year. This balance was the numerical representation of the weighted ideal summer / ideal winter situation. Other representations were also ossible such as WWR and average heat accumulation, Q/A.

Results

Several areas of interests for the company Grontmij were identified. In terms of how the building actually performed, information is still limited. This is due to the fact that the project was stopped before additional comparisons on the initial quasi-steady-state modeling method and extensive simulations have been carried out. Nonetheless, the simulationists argued that the calculation method is much too simple to predict a realistic energy balance of the building, but the method is still more valid than existing rules-of-thumb. However, there have not been any evidence for this claim, therefore more details on the quasi-steady-state method has to be published and reviewed before any valid conclusions can be given of the particular implementation. Nonetheless the fact that simulationist could and did implement their own calculation method during the limited time frame of the early design stage was an important message for both the simulationists and the building designers.

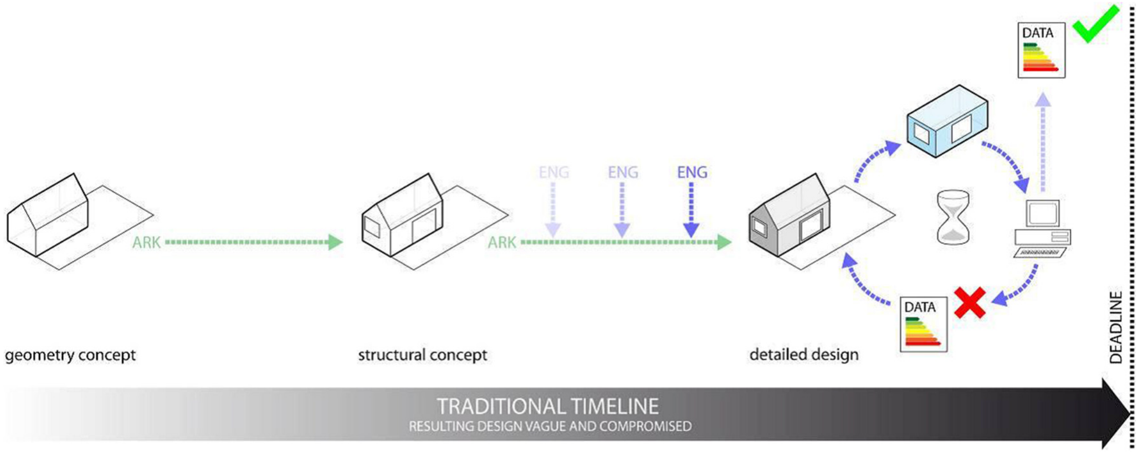


Figure 34 Traditional design approach - consulting engineering teams (simulationists) have limited time to inform the architects (building designers) late in the design process. Image by courtesy of Grontmij ©

The main argument from the simulationists to utilize the Consequence based design approach was to reduce the limitations of the traditional timeline, where evaluations are introduced very late in the design stage. The traditional design approach according to Grontmij is identified as a *late design stage verification process assisted by engineering* (see Figure 34). This usually means that designers develop their concept with limited input from the engineering team (simulationists) and the input the building designers receive is based on rules-of-thumb and simple guidelines. This type of design approach has proven quite robust in terms of collaboration with different studios and their different ways to apply design methods in the early design stage. However, projects like in this presented case, which are geometrically complex and difficult to analyze, require in-depth knowledge of the dynamic systems in play (e.g. energy and indoor environment related systems). Therefore, simple guidelines and rules-of-thumb are insufficient for such projects. It should be noted that in this argumentation, there is no mentioning of High-Performance Buildings, but the need is simply to qualify minimum requirements.

In previous projects of this scale and complexity Grontmij has experienced that the engineering team had to model the building in dedicated BPS tools in very limited time in the very late design process. Often, the provided geometry model was found unfit for import and the model had to be manually adapted to the BPS tool. If or when the engineering team found that the design did not meet the performance criteria (e.g. energy frame calculations), the engineering team had to deliver fast (and often crude) feedback sketches and concepts for redesign to the architects. The result was in these cases often a compromised building design, which vaguely met the imposed performance criteria (regulations).

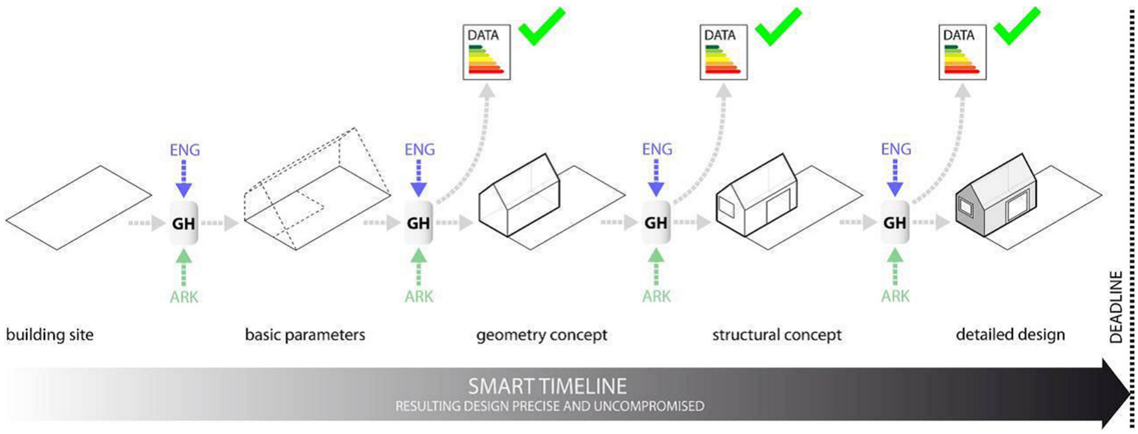


Figure 35 Consequence based design approach extends the timeline by introducing evaluations in the early design stage. Image by courtesy of Grontmij ©

In the Consequence based design approach seen in Figure 35 (at this point called a "smart timeline"), the design team works together to form a model. The building designers and the simulationists define the criteria that the final design should meet. These criteria have been evaluated in the model and communicated into the design tool (see Figure 36). The model is continuously updated throughout the design stages and thus new and altered criteria may be applied.
In this case the model was completely replaced two times. The first change was an improved evaluation method and the second change was due to changing architectural criteria. The building designers wished to have parametric control the vertical distribution of glass in the façade geometry, which resulted in a complete remodeling of the integrated dynamic model. In both cases changes were made in the parametric data model through the VPL.

The result of this approach according to the building designers and the simulationists was found to be compromising the design choices and architectural freedom much less (compared to the traditional approach seen in Figure 34). Another argument from the building designers and the simulationist was that technical demands, criteria and goals were more likely to be met. This can also be seen in the agreements on statement E (Table 4), where both architects and engineers "Agree mostly".

Table 4 Level of agreement to the five statements A-E (see Section 4.1)

		Architects	Desginers	Engineers	Others	Total respondance	
Agree completely	100	90	-	80	-	83	Statement A)
Agree very much	80	50	-	72	-	64	Statement B)
Agree mostly	60	60	-	74	-	69	Statement C)
Disagree mostly	40	25	-	61	-	49	Statement D)
Disagree completely	0	75	-	68	-	70	Statement E)

One response from the survey revealed that the integrated dynamic model in particular affected the discussions regarding design consequences on the architect's part versus energy performance on the engineer's part. But as it can be seen from statement D, the architect(s) did not consider the model to assist in the exploration of architectural expressions and concepts of the building design that were not directly associated with the (simulated) building performance. This probably was due to the very limited amount of flexibility of the model. This claim is confirmed in Table 5, as *flexibility* is ranked highest among the features requested from the model.

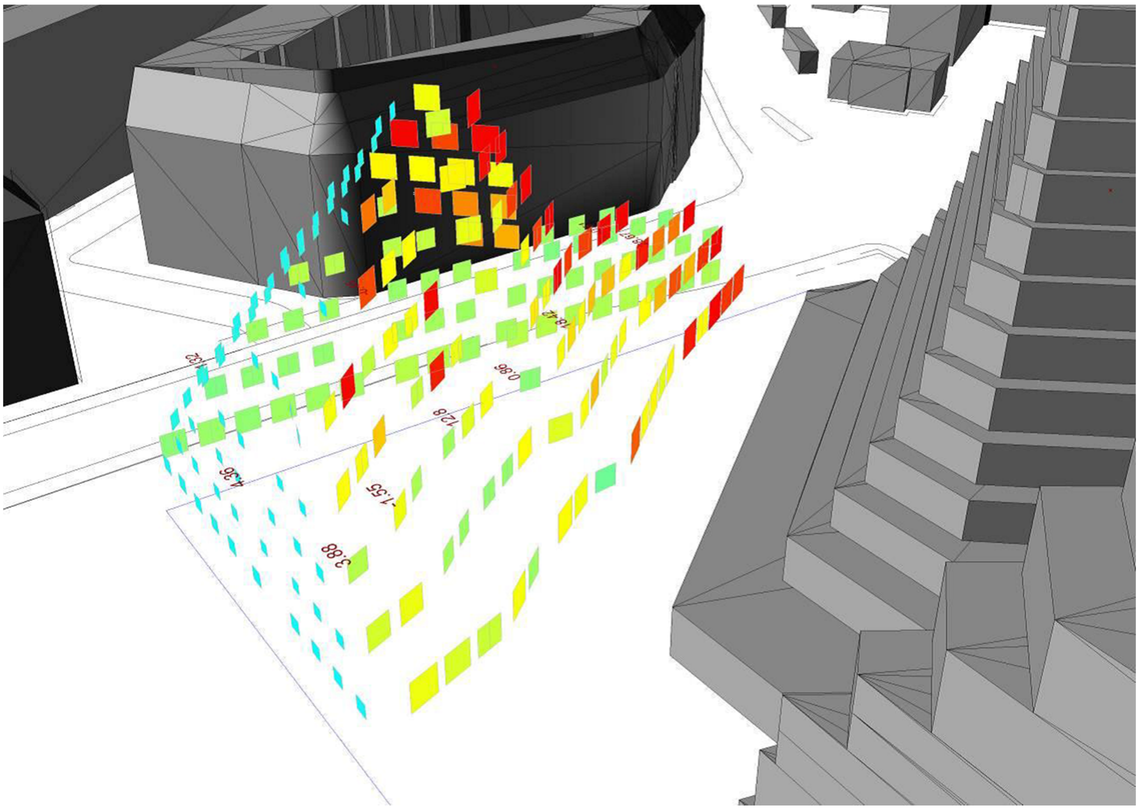


Figure 36 The simple geometric representation of the model revealed to be a powerful communication tool. Image by courtesy of Grontmij ©

Consequence based design seen from the simulationist's point of view

The following benefits and drawbacks of the Consequence based design approach based on this project were identified[20] by Grontmij:

· More precise evaluations
· Improved communication from the simulationists to the building designers
· Faster results delivery
· Reduced worker hours for simulationists

In general, the engineering team found itself to be able to "*perform better calculations, evaluations and design change proposals in a coordinated way in comparison to the traditional approach"*. The engineering team was able to better communicate to the architects in the design team, as the communication was based on the very visual sketch-like models. Grontmij thus identifies itself as a more "*attractive partner in integrated design projects"*, as the engineering team is able to build design evaluations and consequence analyses that are much more project specific. In terms of external activities, the engineering team found itself to be able to "*deliver a larger and more valid basis for design decisioning in the early design stage"*. Specifically, this was found essential in relation to criteria in terms of energy consumption, indoor environment and load bearing structure. The engineering team found that it was "*able to shorten their own sketch-process thus delivering proposals faster with integrated dynamic models"*. Whether these statements are applicable for all consulting engineering firms have not been examined and therefore the results shown here must be seen in the narrow aspect in which it is analyzed.

Internally the company also identified activities that reduced worker hours spent on the project. One example that was highlighted from an internal report is that "*the engineering team does not need to remodel the building for separate tools when integrated dynamic models are used"*. The team also "*reduces the amount of errors caused by parallel tool modeling"*. As a consequence, the number of quality check-ups is reduced as well. Grontmij further argues that the (Consequence based design) approach enables the engineering team to react faster to external design changes. Primarily because the changes are made with the (integrated dynamic) model and this creates awareness throughout the design team. All these findings correspond to the general characters of integrated dynamic models as discussed in Section 2.4.5.

[20] These identifications are primarily taken from internal evaluation documents and an internal business potential report. These documents could unfortunately not be attached to the thesis due to confidentiality, however citations and references were allowed published.

Table 5 Preference towards the model features (see Section 4.1)

		Architects	Desginers	Engineers	Others	Total respondance	
Highest priority	5	5,0	-	4,5	-	4,7	Flexibility
	4	4,0	-	2,5	-	3,0	Speed
	3	1,0	-	3,5	-	2,7	Precision
	2	2,0	-	2,0	-	2,0	Usability
Lowest priority	1	3,0	-	2,5	-	2,7	Visual Quality

Economic expenses and benefits

The economic benefits (of the consulting engineers) gained from the Consequence based design approach are identified as the ability for the consultants to manage challenges that used to be difficult and costly to solve. These benefits are categorized as savings and new opportunities.

In terms of savings, the primary saved worker hours of this project are found in the reduced amount of times the engineering team had to re-evaluate the design proposals. This confirms the “cost of change” recognized by MacLeamy (2013). And unlike this project, in previous projects changing geometry and programs has been found costly for the company.

Grontmij found that integrated dynamic models gave the engineering team the ability to receive and process far more design proposals. And the model allowed them to show the consequences of even small design changes. Grontmij further argues that the “*savings will likely be found in larger projects where there is reason to make several changes”*. In terms of the economic benefits in new opportunities: Consequence based design and integrated dynamic models have according to Grontmij opened a new door that allows the design team to analyze on more complex projects faster. Firstly, the team had the integrated dynamic model to perform “*more complex analyzes”* and secondly, the team was able to “*carry out more precise calculations of more complex geometry”*. This means that the engineering team was able to perform tasks that previously had not been possible for Grontmij. On the cost side, the use of integrated dynamic models in this project is very difficult to value; this is due to the fact that in principle one has to evaluate the cost of tasks *“that previously have not been possible”*. However, Grontmij identified the cost of applying integrated dynamic models in new domains as to the time taken to develop new algorithms and methods. In this case, the cost of developing the heat-balancing method was considered outweighed by Grontmij by the benefits of the application. However, they found that higher quality assurance was needed in the method to evaluate building energy consumption, and the best approach might be not to utilize in house developed calculation methods.

In terms of economic benefits and costs relating to the building designers in the design team, a similar analysis has not been carried out. However, as the simulationists have identified the potential of faster and more reliable and impactful results, it may be established that the building designers may potentially receive more accurate knowledge in terms of High-Performance Buildings than by alternative methods. And all respondents agreed that the use of integrated dynamic models has positively assisted in achieving *higher performing buildings* without compromising the architectural expressions and concepts of the building design (Statement E, Table 5).

Conclusion

The integrated dynamic model in this particular case was handled by the design tool Rhino, the VPL (Grasshopper) and a simple quasi-steady-state BPS tool created by Grontmij. According to Grontmij, the approach of Consequence based design was found to have several benefits over the existing design approaches that Grontmij used to apply in the early design stages. Many of these benefits have an economic incentive, which is aligned with the aim to reduce early design stage changes strived for by e.g. MacLeamy (2013). The current project could not document that the entire design team had the same levels of economic benefits gained through the approach. The survey respondents agreed that Consequence based design approach and the particular integrated dynamic model had improved the chance of creating High-Performance Buildings. This is quite interesting as the original motivation for implementing and using an integrated dynamic model was purely based on less ambitious (minimum) requirements imposed by regulations. What is determined is that this particular project would not have come this far in the design process, if the implementations of the integrated dynamic model had not been available. The model allowed higher levels of access and control of the heat balance of the building at window level, which ensured a higher quality assurance. In addition, it can be concluded that the use of integrated dynamic models in this particular project has created a better common starting point of collaboration between the members in the design team.

4.1.2 Case study: Kirk

This case study was carried out for Lundgaard & Tranberg architects working with Studio Olafur Eliasson. The project was simply a test case combined with the development of the plugin Termite (which is discussed in detail in Section 4.4.3). The idea was to test some of the limits of parametric modelling with integrated dynamic models. The documentation is based on various internal reports, interviews and a survey. As this project has not been designed using a Consequence design approach the results of the survey are only presented here in a qualitative format and only regarding the configuration of the integrated dynamic model. The focus of this case study is to demonstrate some of the model- and tool specific limitations and qualities of integrated dynamic models.

Introduction

The Kirk project is a highly complex masonry building designed by Studio Olafur Eliasson. The project is to be completed in 2016. Grontmij has not participated in the design, nor participated in consulting the building design of the building. However, by a third party Lundgaard & Tranberg, Grontmij was consulting on issues on related BIM-developments on the building project. Specifically Grontmij transferred highly detailed double curvature surfaces from one design tool to another. In this process it was decided to *reverse engineer* the main structure of the geometry model and process this into the parametric tool Grasshopper as a parallel test case. The reason for this was twofold. 1) Grontmij used this as a case study for internal training purposes as reverse engineering would allow more people in the company to work with Consequence based design and 2) at the request of the author, to test the limits of the plugin Termite (Negendahl, 2014c), which was under development at that time.

Modelling highly complex buildings with high level of parametric flexibility

In Section 2.1.4 on the discussion on parametric flexibility, it was specified that sufficiently flexible models will accommodate changes spanning multiple levels of information. And given a high level of flexibility, the parametric model can absorb new information and ongoing decisions. However, in practice Harding et al. (2012) experienced limitations to the parametric tools and in some cases the parametric model schema would not cope with the large scale changes. In this case, the interest was based on how *complexity limited the flexibility of the integrated dynamic model*. Further, the aim was to see to what extent might it be (commercially) viable to consider integrated dynamic models in complex buildings.

The existing model material was composed in mixed design tools of fixed geometry delivered in 3D and in 2D by the building designers. Once the entire geometry model was imported in Rhino, the “reverse engineering” or parametrization of the geometry was initiated. In Figure 37 the parametric base model is seen. The dark intersecting barrel-like features are representations of four building sections. The white “cut-out-pillars” represent the geometry that was subtracted from the building sections. This Boolean intersection created the placements of windows around the building. At this point, any of the many pillars or barrels could easily be shifted, resized and modified in many other ways, thus fully parametric and re-defined through the VPL.

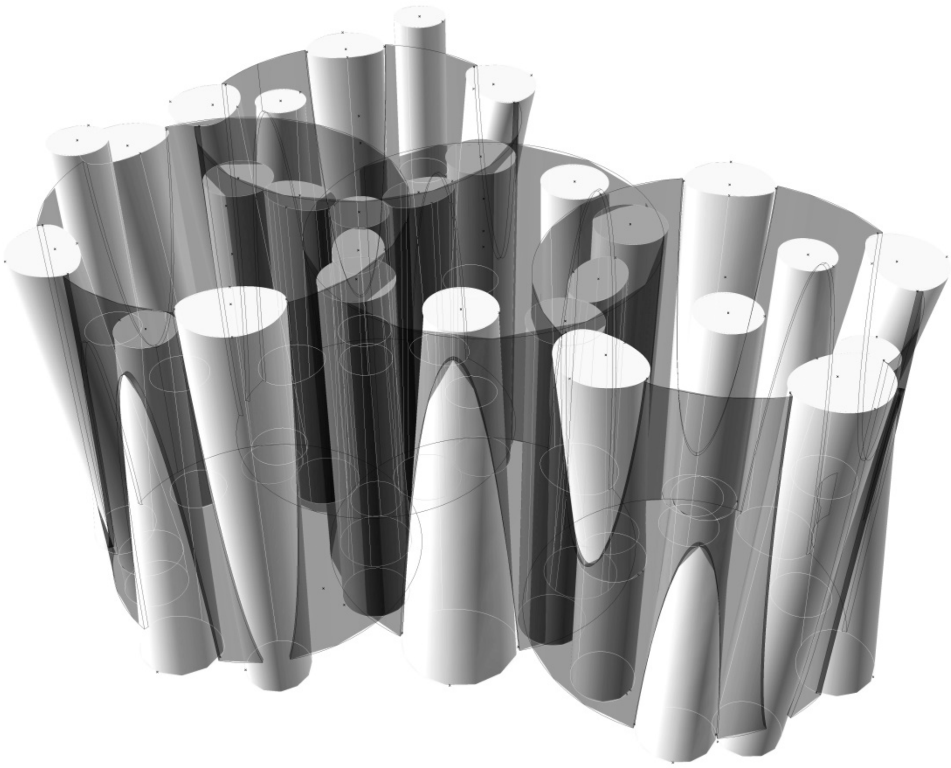


Figure 37 The parametric base model was created by the translation of non-uniform "cut-out-pillars" seen in white, which defined the cut-outs for the glazed areas in the masonry-façade. Image by courtesy of Grontmij ©

The next step was to (parametrically) model the details needed to simulate the energy consumption of the entire building. To do this it was necessary to completely model the building envelope and subdivide the internal space into thermal zones. Additionally, the VPL was used to define a parametric control of HVAC-systems systems and non-geometrical features like internal loads. This new and much more detailed parametric model was coupled to the BPS tool Be10 (SBI, 2013) through an early version of Termite (Negendahl, 2014c) to form an integrated dynamic model (Figure 38).

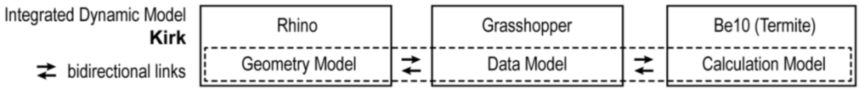


Figure 38 The integrated dynamic model. The design tool Rhino combined with the VPL Grasshopper and Be10 through Termite

Results

The real challenge was to maintain flexibility while subdividing the geometry. Much effort was put into representing the geometry as truly as possible, coming down to the smallest details of the exact curvature of the cut-outs. This led to various challenges in maintaining automatic translation into zones; for example, assigning rules to calculate the inner volume based on the highly varying outer geometry was a challenging task. Inner volumes however could easily be estimated in other ways, but the test was performed on the extreme case, and therefore automations were included in almost any conceivable parametric relationship. As it can be seen in Figure 39 (to the right), the graph-based model was growing out of proportions. Every window could for example be individually tweaked by parameters. It became quite obvious that a different type of parameter tuning was necessary if the simulationists were to have control over the building in future projects.

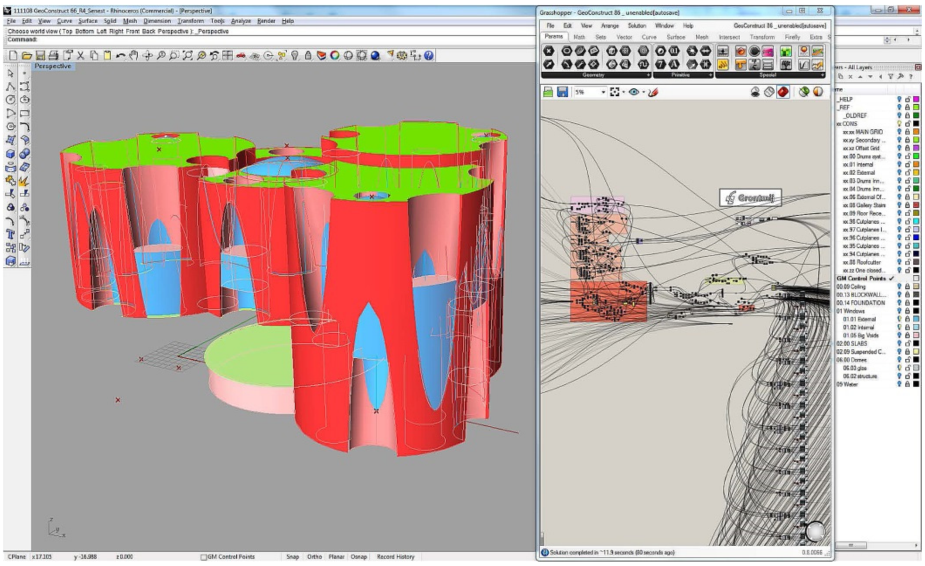


Figure 39 The integrated dynamic model was growing in complexity and scale. To the left are various surface types seen in different colors. Each surface has several ontology rules associated with them e.g. material type, thickness, orientation etc. Image by courtesy of Grontmij ©

The integrated dynamic model was able to represent a precise geometric representation of the building. The geometry model was quite possibly far more detailed than the BPS tool is developed for. In this sense, the case demonstrated exactly what it was supposed to: it is possible to parametrically model a complex building with integrated dynamic models. However, the extensive graph-based script, and the simulationists' devotion to parametrical definition to test every parameter the BPS could tool could handle, made the model very computationally heavy. The simulationists measured a speed drop of a factor 10 from a simulation with very simple non-geometric model to the complex model shown in Figure 39.

This not only made the model harder to deal with, it also meant that the design feedback became much less interesting for the design team. Therefore, fast feedback was found a highly valuable characteristic of the integrated dynamic model.
The very specific ways the pillars had cut the building, the window geometry and zoning had to follow dis-continuous modeling principles. For example, in some situations the cut-out-pillar was creating an arch and in other cases splitting the building volume in two. This required different types of automations. In some combinations of splits and arches the model ended in creating non-compliant geometry, which made the model return an error. This error was propagated into the BPS that returned wrong results. It was found necessary to construct quality check-points in the model to ensure the BPS tool was receiving what it was meant to.

Based on the authors own experiences the following lessons were learned in relation to Consequence based design approach:

- Immediate results are very valuable
- Large graph-based models slow down the consequence feedback
- Integrated dynamic models can improve modeling precision significantly
- High visual quality is valuable for internal communication
- High visual quality is valuable for external communication
- Consequence feedback can be delivered on both parametric and fixed models

The case showed that coupled BPS tools are of increasing interest for the Consequence based design method if feedback is faster. A highly complex model that is computationally expensive takes several minutes to generate feedback, and this was found to be a limiting factor for the design process. Considering the alternative, i.e. manual updates in separated BPS tools that would take hours or even days to update, the approach is however far more relevant for the early stage design process. Precision is important, but the important findings here were that speed is also highly valuable, and if precision was the limiting factor, speed became even more important.

A key-finding was improved precision. When the simulationists compared the calculations made with Termite with existing results made by manual remodeling in Be10, the simulationists found the integrated dynamic model inputs to be between 10 % and 25 % off the manual model. Especially the doubled curved surface areas were greatly under-estimated in the manual model. For example were the curved exterior walls greatly overestimated (in some places 25% larger than in the referenced model). Therefore, it must be concluded that the measurements and direct referencing techniques used in integrated dynamic models improve accuracy significantly over the manual remodeling approach.
The simulationists further identified the visual dimension of the model to be highly valuable. This was found especially valuable for communication purposes to other members of the design team because this level of visual communication previously had not been possible with other tools. Within the engineering team, it

was now possible to communicate the performance requirements in a much more direct fashion. Therefore it was concluded that such models would be of great value when multiple in-house disciplines were to cooperate and jointly communicate design changes to the building designer. In relation to this it was found that the architects working on the particular project were very interested in receiving visual performance feedback over traditional reporting, even though the architects themselves were not in control of the actual model.

Conclusion

From this case study it was found that great planning in constructing the integrated dynamic model is needed ensure valid feedback from the BPS tool. There is a need to develop robust ways to quality check the model before and after simulations. From an analysis of the particular integrated dynamic model, it is established that the traditional way of modeling complex buildings (especially double curved surfaces) in the BPS tool Be10 has been very imprecise. The tool Termite showed that with fixed (non-parametrical) geometrical models, even double-curved surfaces could be effortlessly evaluated through the integrated dynamic model, thus increasing the validity of calculations. The fully parametric integrated dynamic model showed it is possible to model any design variable in the BPS tool. As long as the integrated dynamic model allows the design variable to transmit into the BPS tool, there was no limit in how many open design variables the model could administer. Nonetheless, due to the limitations of the graph-based VPL, the number of open design variables, and the ontology rules that followed to control these, increased the computational load significantly. Still, the integrated dynamic model allowed the simulationists to calculate several very different solutions in few minutes. This would manually have taken (working) hours or even days more than by using the automations made possible by the VPL and Termite. The building designers found the results much more relevant as the model could deliver visual results in turn for deductions in a report.

4.1.3 Case study: Nordfløjen

This case focuses on the façade design of a new hospital in Denmark during the early design stage and concerns the development of a High-Performance hospital Building. Specifically, this includes a discussion of measures taken to maintain high levels of daylight and direct sunlight in each zone, while passively reducing energy consumption.

Rules-of-thumb, pre-simulated data and entire building energy simulations in Termite (Negendahl, 2014c) are combined into an integrated dynamic model. This case study documents a way to manage opposing criteria and control heat and energy balances with integrated dynamic models. The documentation is based on various internal reports, the competition entry, interviews and surveys. The team behind the project consisted of 3XN, Aarhus Arkitekterne, Kristine Jensens Tegnestue og Nickl & Partner Architekten AG and Grontmij Architectural Engineering.

Introduction

In the past few years, numerous hospitals and large scale extensions of existing hospitals have been designed and are now being built in Denmark. The new hospitals are part of a nationwide plan to improve the efficiency of the Danish healthcare system. One of these new hospitals is a 52.500 m^2 extension of Rigshospitalet in central Copenhagen. The new extension is called *Nordfløjen,* in English "*The North Wing*". The project is to be completed in 2017.

A three step competition to the general AEC-industry was issued in 2011 (http://www.regionh.dk/, 2011). One of the design requirements was to utilize an *evidence based design process* (http://www.rigshospitalet.dk/, 2010), and thereby documenting that the design proposal would benefit faster recovery of patients. Research such as R.S. Ulrich, C. Zimring et al. (Ulrich et al., 2008) concerning the implementation and assessment of Evidence-Based Design (EBD) of hospitals emphasizes the importance of using credible data in order to influence the early design process. Therefore, from a very early design stage, the simulationists were invited to increase the knowledgebase of high quality indoor environments to the rest of the design team. Another design requirement was a very low building energy consumption matching the Danish 2015 energy regulations (Danish Building Regulations, 2013) of office buildings. This meant that the focus of high quality indoor environments was greatly challenged by reduced energy consumption.

Analysis of the winning proposal

Some of the main winning conditions based on the judges' feedbacks were that the final proposal of Nordfløjen had a well-organized building layout to minimize walking distances for staff. The planned functions requiring natural daylight and sunlight were placed in the *best* areas possible. In this way it was possible to increase staff-to-patient contact, and higher sunlight and daylight exposures would increase patient recovery rates.

To achieve these winning conditions the solar loads on the building envelope had to be optimized along with the internal planning. This required a high demand of sun exposure to certain room and area functions, which again required precise handling of the façade and ventilation, and cooling system. It also required a model that was able to remap the entire functional space planning in a very fast and efficient way. The proposal turned out to be one of the most cost efficient projects among the competing proposals, both in terms of construction costs and life cycle cost.

The central argument in the integrated design methodology (Löhnert et al., 2003) is that investing worker hours in evaluating building performance is always paid back in terms of the total life cost of the building. The problem of participating in competitions is the little incentive of investing in such upfront expenses when there is no guarantee of winning the project. Nevertheless, the project was handled as an integrated design process as it was believed by the design team that low life cycle cost was one of the important winning criteria for the client.

Handling design criteria

The main task from the very early design stage was to ensure that all participants in the design team had a completely clear picture of all design criteria and requirements. In large scale projects with several stakeholders interacting, such a task can be very difficult. Many of the criteria, even governmental requirements, were up for discussion at least once. The way criteria and requirements shift and change during the early design stage can be very expensive in terms of worker hours if the altering building proposal is to be evaluated again and again.

The various members of the design team had biased and separated focuses from the start; for example, the simulationists had focus on balancing indoor environment and energy consumption. One sub group of the building designers (Plan team) had a crucial understanding of planning the layout efficiently and had their main focus on this part. Another group of building designers (Façade team) focused their work on the façade as well as the inner and outer appearance and experience of the building. The simulationists in the team were divided into two groups. One group (Structural team) focused on bringing costs down by rationalizing geometry in terms of static considerations and handling fire escape routes etc. The other group (Energy team) primarily discussed in this case study, handled the building performance in terms of energy and the indoor environment (daylight, sunlight and air quality). All members were specialists in their respective field, but very few had an insight in all the design requirements and end goals for the project as a whole.

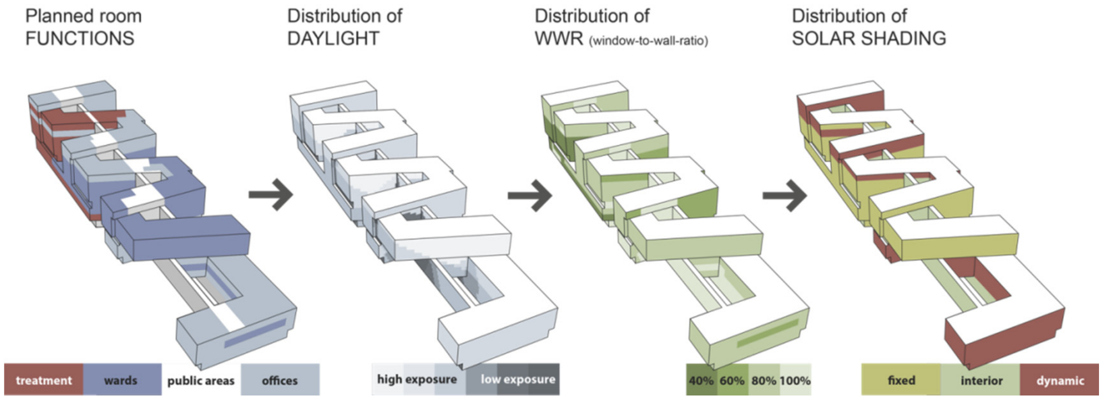


Figure 40 Overview of the outcome of the integrated dynamic model of Nordfløjen. The model generated continuous energy performance feedback that again affected the planning of room functions, WWR and placement of solar shadings. Image by courtesy of Grontmij ©

Embedding design criteria in an integrated dynamic model

It was decided amongst the Façade team and the Energy team to use an integrated dynamic model to simulate and visualize the "optimal" WWR and placement of solar shading on the façade. This would help the simulationists understand the architectural concepts as well as providing feedback of building energy performance, daylight and sunlight performance to the architect in their native design tool (Rhino). The Façade team was able to communicate what they found to be optimal in terms of WWR and placements of solar shading to the simulationists who in turn attempted to balance out the competing criteria of energy consumption and indoor environment.

For the integrated dynamic model to work as intended, various requirements and end goals had to be correlated between the Façade team and Energy team. The teams collectively decided the rules of minimum requirements of energy consumption and indoor environment as well as the architectural goals of a *diverse façade* and a green roof. The diverse façade was a way for the Façade team to explain how opaque and transparent surfaces in the building envelope had to appear differently from any angle.

Four main design criteria for the building envelope were formulated:

- Green roof strategy, minimize the need of solar panels
- Diverse façade strategy, ability to generate window distribution to appear handcrafted and ability to fit the zone planning in different ways
- Daylight strategy, maximize the daylight conditions in wards, public areas and offices.
- Sunlight strategy, maximize direct solar exposure in wards

A green roof strategy meant that large parts of the roof area were to be cleared of solar panels, which again meant little to no electrical contribution to the electrical energy consumption. The green roof strategy combined with the need of a very transparent façade meant that the electricity demand for ventilating and cooling the building had to be controlled, since the building had to live up to the Danish 2015 energy requirements (Danish Building Regulations, 2013). It was important to control the energy properties of the building envelope in detail. Therefore, many different suggestions of controlling the overall building heat balance were discussed. The most cost efficient system was to utilize a district heating and cooling system and to minimize the need of external dynamic solar shading.

The main energy and indoor requirements were formulated:
- Maximum overall building energy consumption of 41.1 kWh/m^2 per year
- Minimize the need of solar shading systems
- Maximize the window wall ratio of the façade

Setting up rules of an integrated dynamic model

An integrated dynamic model was set up by the Façade team and the Energy team collectively; the model rules were controlled mainly by the Energy team and refined during the design process. The model worked by dynamically reading geometric data from the fixed architectural geometry model made in Rhino.

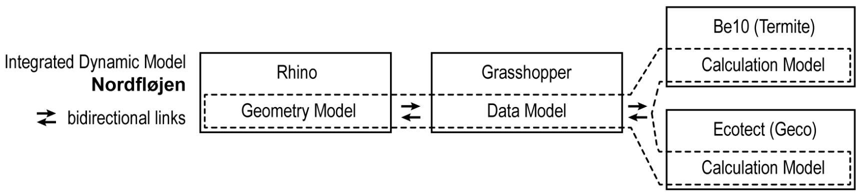


Figure 41 The integrated dynamic model. The design tool Rhino combined with the VPL Grasshopper and the BPSs Be10 (Termite) and Ecotect (Geco)

To ensure that the data model was properly formatted when sent into the coupled BPS-tools (Ecotect (Autodesk, 2013b) and Be10 (SBI, 2013), see Figure 41) the Façade team and the Energy team came up with a simple rule set of data handling (much like a simple MVD, see Section 2.4.5), which mainly consisted in using explicit Rhino layers for drawing predefined objects. One layer, for example, could only contain roof geometry.
To maintain maximum control of the indoor environment performances of the various room types, the Façade team separated the façade into six different layers corresponding to six different categories (in Figure 40 to the left only four of the categories are illustrated).

- Public areas (e.g. hallways)
- Technical areas (e.g. server rooms)
- Treatment areas (e.g. operation rooms)
- Ward areas (e.g. wards)
- Office areas (e.g. reception)
- Closed façade (e.g. in front of lifts where no windows are needed)

The idea was that the different categories had different criteria of indoor environment (i.e. thermal requirements, air quality and daylight distribution). Instead of controlling each room individually, the categorization helped the team to control the whole building in the integrated dynamic model. Categorization by room type is however not sufficient to take the highly irregular weather effects and solar distributions into account. For that reason, three further subdividing categorizations were developed as illustrated in Figure 42.

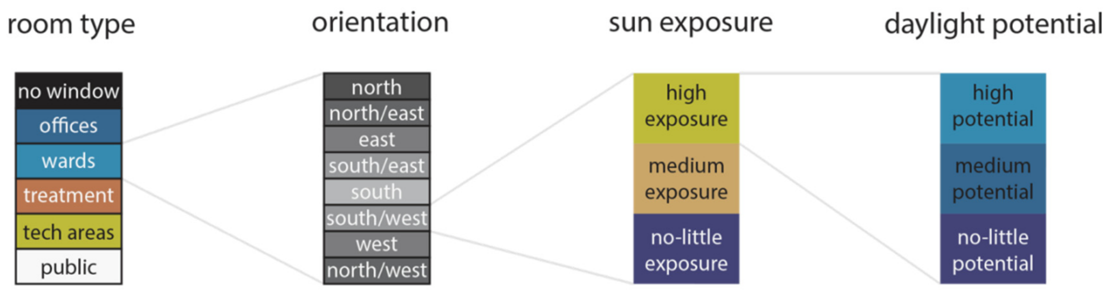


Figure 42 complete characterization scheme: room type, orientation, sun exposure and daylight potential in that order. Here the daylight potential of wards facing south/west in high sun exposed areas is shown.

The integrated dynamic model was set up with two different BPS tools. Ecotect was used to calculate the average direct solar exposure on the façade as well as the access to daylight by calculating the ambient daylight on the façade. The second BPS was the Danish national building energy calculation software Be10 that was utilized to perform a whole building energy simulation. The VPL Grasshopper was used to link the BPS tools to Rhino. The link to Ecotect was handled by Geco (UTO, 2013) and the link to Be10 was an early version of Termite (Negendahl, 2014c).

The results from Ecotect were read back into the model (see

Figure 43) and used to sort the façade into three different “sun exposure” categories (Figure 42). Essentially, the sorting mechanism grouped the façades with a high degree of solar insulation, to help estimate where external solar shading would be most efficient. Ecotect was also used to make another simulation of daylight potential (CIE-overcast sky simulation measured in lumen). This simulation was used to evaluate each square metre façade upon its access to daylight; “daylight potential” (Figure 42). These subdivisions were thus further separating the façade into three sub-categories; *no-to-little daylight potential, medium daylight potential* and *high daylight potential*. Essentially, the daylight potential was an evaluation of shading magnitude of the surroundings and the building itself. The whole categorizing scheme resulted in individual control of 6x8x3x3 = 432 different ways to design the façade. Based on the outcome of these sub categories, the most effective choices of glass-types, along with different window properties were chosen by the Energy team (aided by various automations and design rules as explained below) and sent as an input to the second BPS tool Be10.

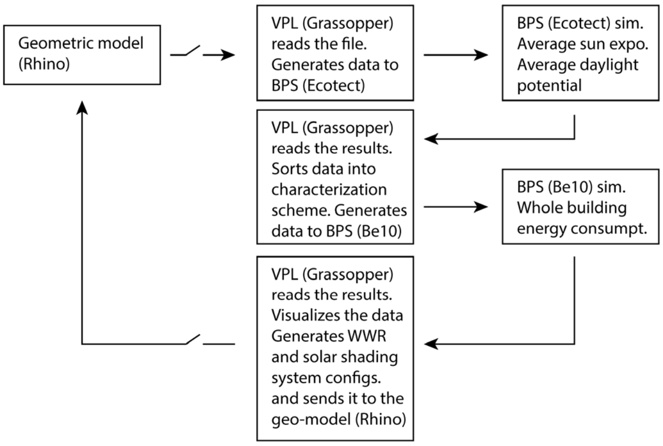


Figure 43 Process of using the integrated dynamic model across members of the design team. The Geometry model defined by the Façade team was read by Grasshopper. Ecotect was used to make two calculations a) a solar insulation simulation and b) CIE overcast sky simulation. Grasshopper was used to sort the façades into a characterization scheme which was fed by Termite into Be10. Be10 was used to simulate the overall energy consumption of the building. The results were sent back to the Façade team with WWR visualized in the original Geometric model.

Each of the 432 options in the categorization scheme could essentially have any number of design variables attached to them, for example individual control of glazing type and window height. This of course is very time consuming if automation was not an option. The VPL made it easy to implement various design rules. An example of such a rule is "for zones with the need of high solar exposure, adjust the g-value by the variable α in each of the windows in the façade". This could be rewritten into a g-value dependent on the average solar insulation on the façade, Q_{sol} (W/m^2) and the principal variable α:

$$g = Q_{sol} \cdot \alpha$$

This and other similar rules made it possible to control the 432 "unique" façades by 48 design variables much like α. These design variables covered generative rules on WWRs, rules to set enumerations on thermal conductivities, rules of setting of solar shading types, as well means to adjust as electrical lighting control systems, cooling loads and solar cell properties. A few of the 48 design variables were used to adjust the relative magnitudes of sun exposure and the daylight potential. This was done manually as the three way sub-division was found to be too simple to represent every single aspect of the design.

Results

The Energy team chose to visualize the effect of WWR in the Façade team's model as colors on the façade as seen in Figure 44, but the actual results were represented as lists containing WWR, window type and properties and solar shading type. These lists were generated as an alternative to generate actual façades on the building (principle illustrated in Figure 45) for one reason: the Façade team wanted to manually control the façade composition. The Façade team thus used the data model as a basis for their detailed façade design.

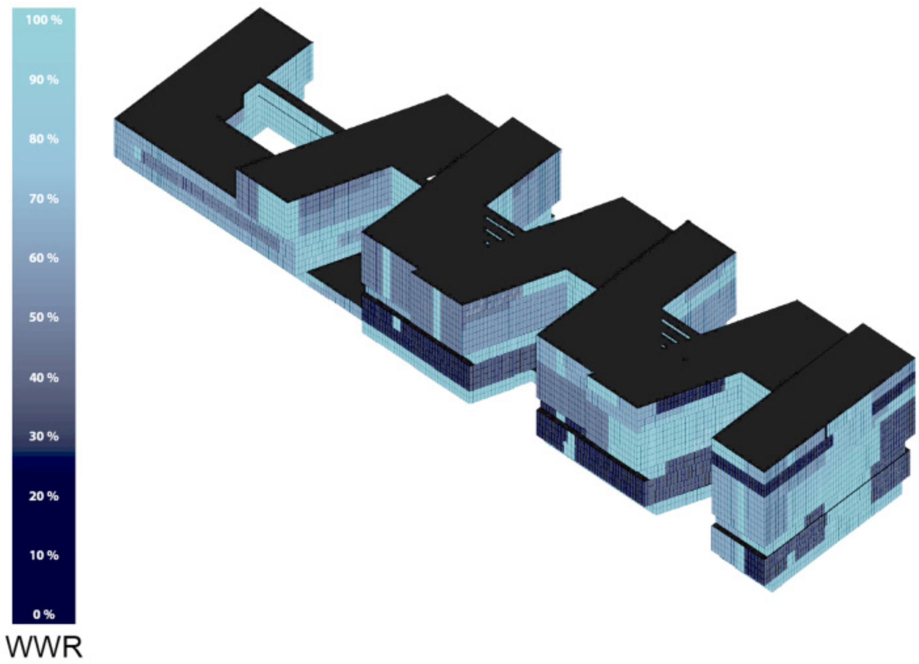


Figure 44 Screen shot of the model as seen in the Façade team's tool Rhino. The colors represent different variations of WWR results from numerous iterations through an entire building energy performance simulation tool. Image by courtesy of Grontmij ©

Several hundreds of different combinations of variables were tested by the Energy team before the chosen results were sent to the Façade team. Each test took less than a minute thus making it feasible to test less conventional design concepts. This process saved many discussions of sub optimality and time consuming adjustments between the two teams. The process allowed the Energy team to seek a solution that meets all the predefined design requirements, and then present the solution via the integrated dynamic model.

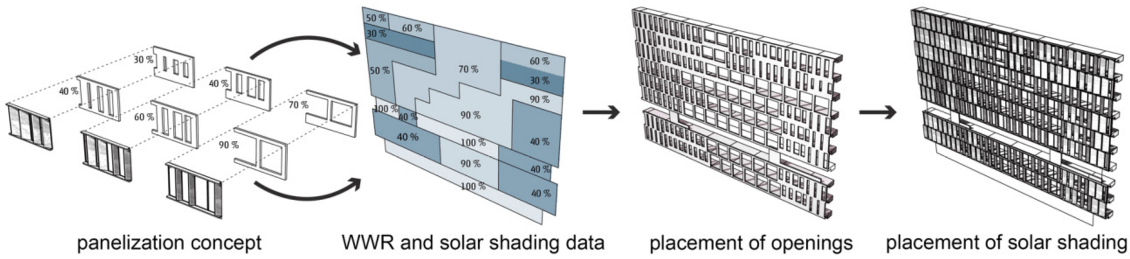


Figure 45 The Façade team used the WWR data to panelize the façade and the solar shading data to place fixed and dynamic shading systems. All data was streamed through a dedicated server linked to the integrated dynamic model. Image by courtesy of Grontmij ©

During the design process changes to the façade geometry and re-planning of the floors were followed by continuous updates of the integrated dynamic model. The model gradually took more details into consideration. For example, the Façade team resolved various design challenges with the composition of the façade panels, and the Energy team settled the details of ventilation systems and the heating and cooling system. At a point in the process it was decided to place parapets at every floor above ground floor, and later the ceiling was retracted from the façade two meters into the room to let in more daylight. Specific panel distributions were tested with the same model and the consequence was continuously monitored by the by the simulationist and the building designers. The integrated dynamic model was used to simulate and qualify several thousands of solutions in this early design stage.

All survey respondents agreed that the integrated dynamic model had created a better common starting point of collaboration between the members in the design team (statement A, Table 6).

Table 6 Level of agreement to the five statements A-E (see Section 4.1)

		Architects	Desginers	Engineers	Others	Total respondance	
Agree completely	100	80	100	94	-	**90**	**Statement A)**
Agree very much	80	60	95	83	-	**78**	**Statement B)**
Agree mostly	60	75	85	89	-	**84**	**Statement C)**
Disagree mostly	40	75	50	50	-	**59**	**Statement D)**
Disagree completely	0	70	95	87	-	**83**	**Statement E)**

Several of the engineers in the Energy team argued that the option to visualize results in the building designers' original model improved the collaboration between the architects (Façade team) and the engineers (Energy team). This is also seen in the engineers' preference toward the visual quality of the model (Table 7). Integrated dynamic models thus provide a flexible way of displaying building energy performance and daylight feedback to the building designers in the early design stages.

All respondents agreed to some extent to the five statements seen in Section 4.1.1 only statement D, concerning "exploration of architectural expressions and concepts of the building design that is not directly associated with the (simulated) building performance", remained unanswered by most survey respondents.

Table 7 Preference towards the model features (see Section 4.1)

		Architects	Desginers	Engineers	Others	Total respondance	
Highest priority	5	5,0	5,0	3,7	-	4,3	Flexibility
	4	4,0	4,0	4,3	-	4,2	Speed
	3	1,0	3,0	1,3	-	1,5	Precision
	2	2,5	1,0	1,7	-	1,8	Usability
Lowest priority	1	2,5	2,0	4,0	-	3,2	Visual Quality

The Consequence based design approach was found to be highly flexible and able to incorporate criteria from many disciplines. The integrated dynamic model was found robust and highly flexible, even though the model was decomposed and processed through multiple levels of design tools, VPLs and BPS tools. One respondent from the survey noted that the impact on the simulated performance could not have done without parametric tools due to the complexity of the design. And the *window sizes and distributions were a direct result of the energy calculations*.

Main challenges associated with the model

It was decided to reduce the need of solar panels and create constraints on room heights in the earliest formulations of the design criteria. If such agreements had not been settled upfront the façade design could have ended up very differently. Criteria happen to change during the design process and some changes may be difficult to implement in an integrated dynamic model without re-scripting the model. It is for this reason essential to stay focused on the design criteria during the entire design process. And if the design criteria changes, all members involved in the integrated dynamic model must agree on the new criteria terms.

The building could have been optimized (by coupling the model to an optimization algorithm, see Section 3.4) in relation to building energy and life-cycle costs, if the teams had wished to do so. However, as the building was designed as part of a combined heuristic design process, optimality was too challenging to process, measure and automate. Nonetheless, the integrated dynamic model was found to provide a combination of qualitative analyses and quantitative analyses in the same model. And as one response from the survey argued, the approach enabled the design team to *optimize solutions,* thus the approach was a clear improvement in relation to the *standard or old-fashioned design processes*. The coupling method using Termite and Geco was found to provide valid feedback on energy consumption and daylight considerations for the early design stage, even though the used method was a simple approximation of the whole building energy consumption and daylight potential. These key performance parameters are only a few of many important performances in High-Performance Buildings the design team need to take into consideration. Other building performance feedbacks e.g. structural, acoustical, environmental, etc. may very likely have had more influence on the final façade design if these considerations had been included in the model. And in terms of daylight, more precise daylight factor simulations of each room would have provided much more valid results. The model simply evaluated the thermal indoor environment based on the average solar gains combined with the use categories (room type). Based on these data the Energy Team had reasoned the amount of needed ventilation, set points and other controls for the ventilation system. A coupled dynamic BPS tool (ideally multi zone model) to evaluate thermal indoor environment might have changed the final outcome. As for the (missing/desired) thermal evaluations the reason that daylight simulations were not part of the model was that the simulations were considered too computationally heavy and too slow to provide the level of responsiveness required by the design team. Also the low information level at this point of modeling meant that crucial information on room and window geometry was missing. As a consequence, it was found very difficult to perform valid daylight simulations and hourly-based thermal indoor environment simulations at this given time in the design process.

Conclusion

The integrated dynamic model gave the simulationists (in this case, the Energy team) the opportunity to simulate the entire building energy performance of a large complex building faster and with higher precision in the early design process than it was previously possible. The interactions between members of the design team were greatly affected by the use of integrated dynamic models. The responses from the survey identified that the design team developed a *common understanding* of what the integrated dynamic model could bring to the design process. Therefore, the effort to develop/implement/use the model was driven by a collective aspiration. This understanding carried a respect for the different architectural and engineering disciplines. One respondent mentioned that the integrated dynamic model *provided a communal “language" that "translated" (engineers’) analysis into tangible design impact* for the rest of the design team. And in particular, the design process advanced because of the accelerated analysis speed due to the approach. The integrated dynamic model and the Consequence based design approach were thus able to accelerate design changes towards High-Performance Buildings. Based on the survey all respondents agreed upon the five statements seen in Section 4.1.1.

4.1.4 Case study: Nordic Built

Nordic Built compared to the other projects had the largest ambition in handling diverse types of building performance criteria. In this sense it sought the widest inclusion of High-Performance Building objectives. The need for fast and dynamic evaluations of material choices, life cycle costs, daylight conditions, thermal indoor environment, acoustical environment and various structural solutions to renovation as well as social implications of these choices were of great importance. To test the limits of Consequence based design it was decided to replace the typical BPS tool with a new type of evaluation tool that could assist the design team in exploring and finding *High-Performance* solutions in the field of social, environmental and economical sustainability. This case study demonstrates that a BPS in integrated dynamic models can be substituted with any type of evaluation mechanism (see Figure 46). Secondly a design team utilizing an integrated dynamic model may use it to evaluate multivariate and quality defined objectives and solutions (illustrated in Figure 47).

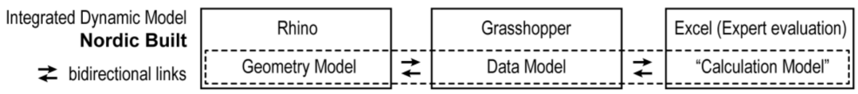


Figure 46 The integrated dynamic model. The design tool Rhino combined with the VPL Grasshopper and the "BPS" Excel relying on expert evaluations within various disciplines

Introduction

Nordic Built was a project carried out with a large integrated design team. The team consisted of Kant architects, Tredje Natur, Niels Bjørn and Grontmij. The project lost in 2nd place in a competition held by Nordic Built. These disciplines covered architecture, landscape design, social sustainability, acoustical engineering, structural engineering, energy and indoor environment. The documentations are primarily extracted from the competition entry for the particular project, supported by discussions, interviews and surveys.

The integrated dynamic model during the entire project was identified as a tool. Unlike the general definition of an integrated dynamic model this coupled BPS tool was based on simple spreadsheet calculations assessed by experts in various fields of disciplines. Therefore, no simulations have been performed by the model. The geometric model was created in different design tools (Sketch-up, Revit and Rhino). In the beginning of the project it was attempted to link the various design tools by using different import/export techniques. This was found too time-consuming as the many interoperability problems took much focus from the actual important task; to design (High-Performance) buildings. Therefore it was decided to develop the integrated dynamic model as a more abstract model, which did not contain any geometry, but only references to particular solutions and the performance of these solutions. Design evaluations would be carried out by all members in the same model through the spreadsheet interface Excel. Excel did not need an introduction and was easy to use by all members of the design team. Separate evaluations (including different BPS tools) were used to inform individual members in the design team.

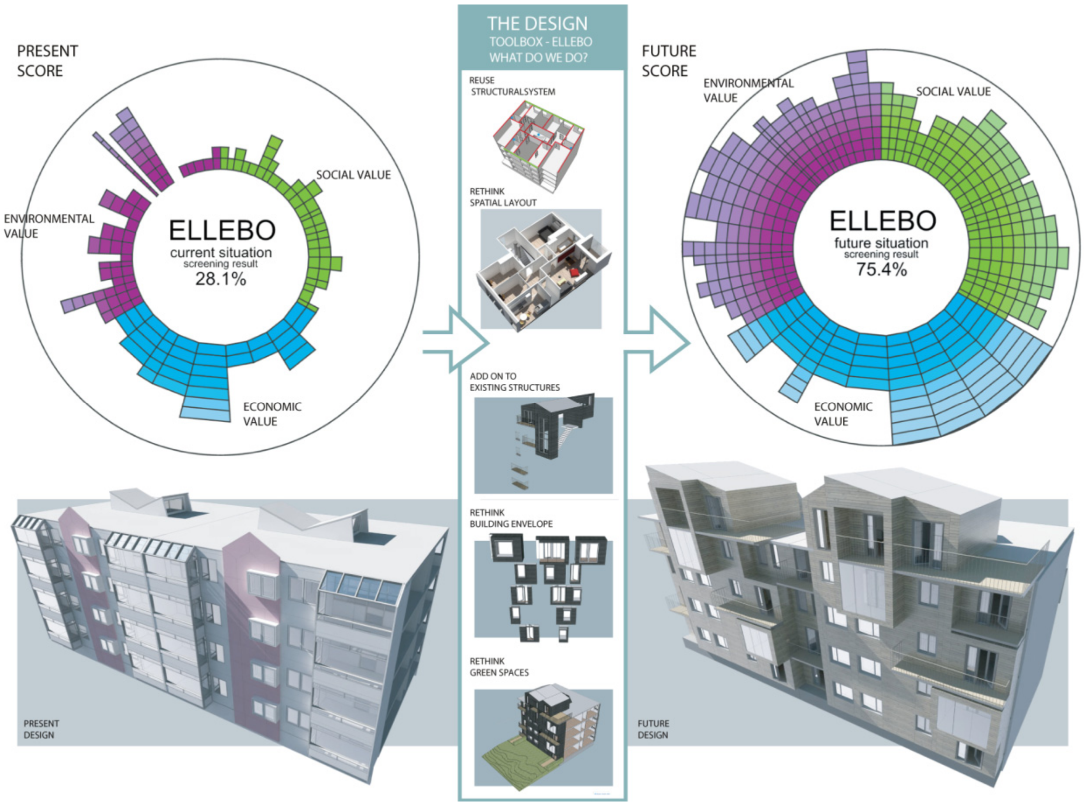


Figure 47 The integrated dynamic model was used as a simple multivariate evaluation tool. Here an example of the scoring mechanisms. Image by courtesy of Kant and Grontmij ©

Combining qualitative and quantitative evaluations in the model

Although few regulations on social performance exist in the Danish building code, social value of a renovation project was considered an important criterion for the design team.

To increase social value was per definition regarded equally important as reducing cost and energy consumption from an early point in the design process. Therefore, to include social sustainability as *just another performance* the team needed to find or develop a fast and robust method to collect and compare social sustainability with any other performance requirement.

As the design team spanned over several disciplines, the value in gathering knowledge of each field of expertise and combining the elements was found the most important task for the design team, but also the most challenging. The

design team immediately recognized the “tool-capabilities” of the integrated dynamic model as means to clarify this process and to compare very different types of performance in a visual and fast way.
By mapping the preservation and renovation options the main goal was to increase the *overall value* of the building. The difficult task when renovating is to find the balance between what is worth preserving and where new additions are relevant. The model helped clarify the relevance of options and when an option becomes a necessity (Figure 48). The design team argued that the model helped them to “*define built environment with much greater human considerations in mind*”.

The design team agreed to design a building that was to maximize the value within the three main categories, which was considered equally important:

- Environmental value
- Social value
- Economic value

The criteria and building regulations

One thing is to carry out a renovation project that prioritizes solutions based on the ability to comply with the building regulations; another is to go beyond regulations. By defining a new design codex, the design team must still comply with building regulations, but it became less important as an end goal. As an example, the design team valued solutions where energy consumption was *reduced* higher than the solutions where energy needs to be *produced*. In the Danish building regulations, the energy requirements do not distinguish between energy created from e.g. solar cells and energy reduced by e.g. more effective insulation. Another example where the building code is insufficient is in addressing resident behavior in terms of energy consumption and indoor environment performance. The design team, for example, valued distributed ventilation control and *live monitoring* of apartment energy consumption higher than a simple but more efficient CAV system. The model showed new levels of insight in energy reduction options. And since most BPS tools are rarely used to model user behavior combined with appliances (e.g. TVs, fridges, etc.) for design purposes, the simple evaluation method drove the simulationists to look into new ways of considering and reducing energy consumption. Again, building regulations impose ways for practitioners of evaluating building performance, but this does not mean that the imposed methods are sufficient for doing so. The distributed ventilation control (here defined as zone level flow control by a BMS-system) was however found difficult to model in the current BPS tools available today. This is mainly because of the very high complexity of user behavior creating high levels of uncertainty. Therefore, when distributed ventilation was used in a solution, the performance was simply estimated by the simulationists.

Figure 48 By defining sustainability, it was straightforward to start the discussion on what is more "sustainable". The "tool" is basically an integrated dynamic model where the BPS tools are replaced with qualitative expert assessments. Image by courtesy of Kant and Grontmij ©

The grading mechanism

The grading mechanism was based on a two-step evaluation procedure.

Each main category (Environmental value, Social value, Economic value) was decomposed by the design team into several sub categories as seen in Figure 49. The relevance of the sub category is rated with a number between 1 and 5. The "relevance factor" is a measurement that indicates the importance of the sub category relative to the other sub categories in the same main category. Therefore t when first established, the relevance can be considered as a fingerprint of the particular needs of the project. The relevance factor is translated into a percentage based on the weight of 1-5 and the relative weight in its main category.

Second part of the procedure is the evaluation of solutions. The design team, as described earlier, produced several solutions in different design tools and BPS tools. Each of these solutions was given a performance score between 0 and 10 in each sub category. The “performance factor” is a measurement of the performance of the sub category. A high performance score indicates a better solution than a low score. The total score is the total weighted performance. A renovation project that performs better than 50% is considered a well performing solution. A performance above 65% is considered an excellent solution.
The way the sub categories are defined and the fact that economic sustainability makes up one third of the total evaluation means that a total score beyond 75% is very difficult to obtain. The reason is that good solutions are rarely cheap, and “perfect” solutions may not exist.

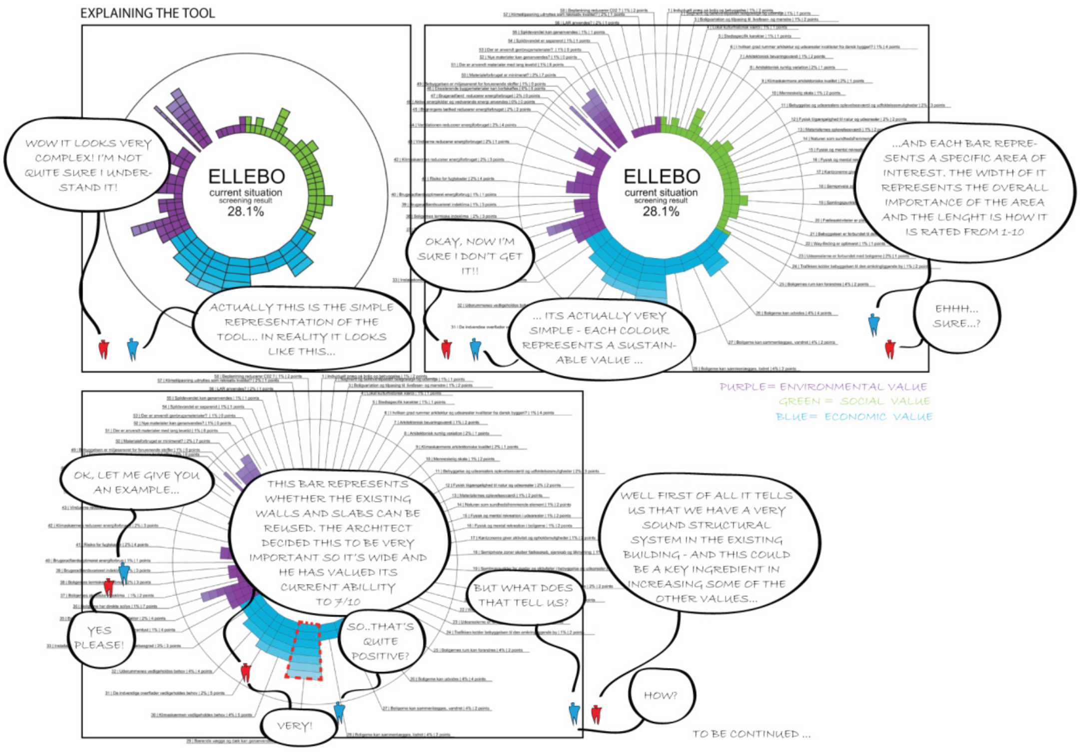


Figure 49 Explanation of the “tool” part 1. Image by courtesy of Kant and Grontmij ©

The model was first and foremost capable of visualizing consequences of the design team priorities. The overall focus is given to the three areas of sustainability: Social value, economic value and environmental value. Each category is given an importance factor of 33% - thus reflecting that social quality, economy and environmental value are equally important. Each of the three categories is divided in a number of sub categories.

The sub categories serve as questions to the performance of the given building. A sub category in the environmental value could be “to what extent are new materials recyclable?” or “to what extent do windows reduce energy consumption in the building?” A sub category in the social category could be “to what extent are the materials experienced as value?” or “to what extend is the residential area connected to the surrounding community?”

As the sub categories could be rated unequally important for the particular site, it created challenges for the experts in each of their own domain (see Figure 50). As an example, (the reduction of) energy consumption through the building envelope was evaluated slightly over the (acceptable) daylight factor in all areas. The design team acknowledged that this prioritization was a challenging task but argued that this was paramount when searching for a better shared and single solution.

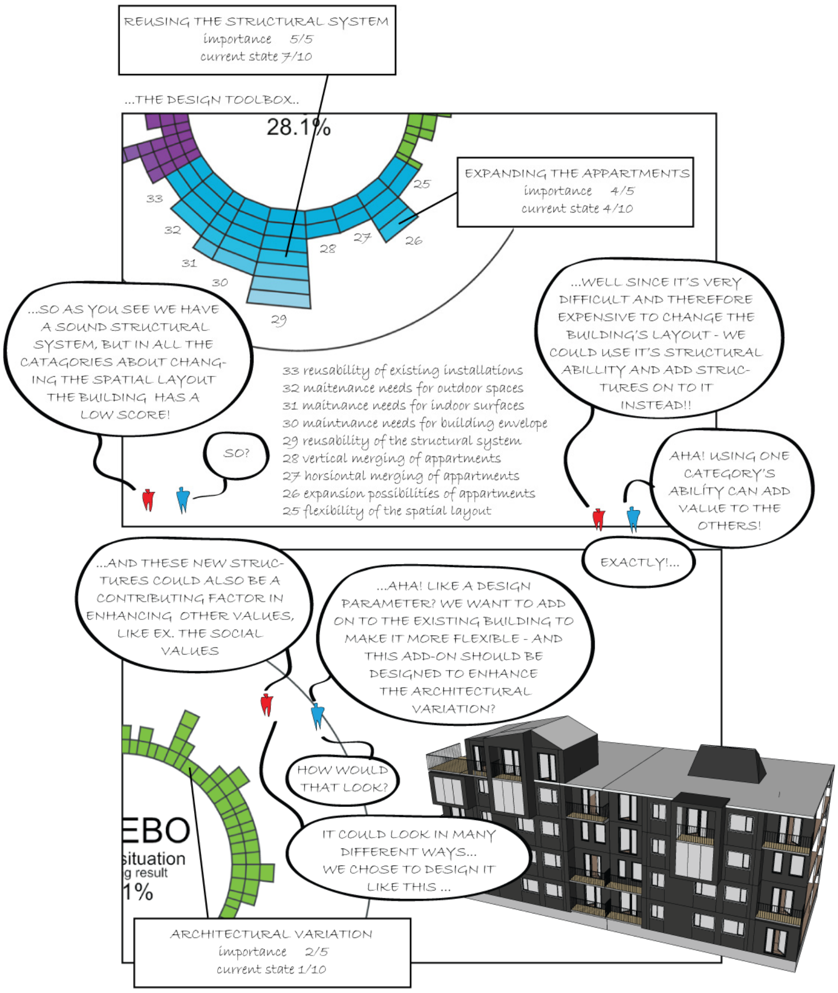


Figure 50 Explanation of the “tool” part 2. Image by courtesy of Kant and Grontmij ©

Results

The design team identified the model as a tool helping the design team (and the client) to choose and prioritize the *most sustainable* solutions, including:

- The model acted as an interdisciplinary facilitation and collaboration tool that helps determine relevance and efficiency of various criteria based on expert knowledge
- The model acted as an analysis tool that helps a client to choose between many options that have different consequences in terms of environmental, economic and social sustainability
- The model acted as an evaluation tool that provides the client with data and knowledge of the impact of the selected choices, before, during and after the building is built

The model served as a dynamic analysis tool during the entire design process. It enabled the design team to discuss and evaluate every option relevant to the different disciplines. The design team argued that the use of an abstract, simple and qualitative model (opposed to a fully automated model working in multiple BPS tools) was much faster and a more direct way of producing solutions of high quality across the disciplines. The design team also argued that the qualitative analysis also facilitated innovation within the design team. For example to utilize a BMS system, well knowing that Danish regulations would not “credit” such initiatives, was innovative.

The choice of an abstract model meant that the model did little to improve the simulated performance and did not aid in the exploration of design as seen in Table 8, statement C. This is mainly due to the model’s inability to host parametric geometry and other design variables. The model could only facilitate the grading mechanism and solutions as reference to existing choices. Nonetheless, the engineers in the team still seem to regard the model as being able to improve building performance. This is mainly because the approach imposed diverse perspectives on the indoor climate concept, which enhanced the reflections of the entire design team. The model therefore helped generating a complex sustainability strategy that included the use of simulated performance evaluations from external tools.

Table 8 Level of agreement to the five statements A-E (see Section 4.1)

		Architects	Desginers	Engineers	Others	Total respondance	
Agree completely	100	50	100	85	-	71	Statement A)
Agree very much	80	50	65	68	-	64	Statement B)
Agree mostly	60	50	2	80	-	46	Statement C)
Disagree mostly	40	50	0	50	-	38	Statement D)
Disagree completely	0	50	100	69	-	67	Statement E)

One respondent from the survey noted that the model was inspiring when working creatively with performance based demands from the pre-design stage. And it directed the programming of functions and helped to implement and communicate complex sustainability concepts. The collaboration goes further than the mutual respect of the professions as the model made the decision of one solution compared to another much more transparent. As the solutions have eminent impact on every discipline, the model made it easy to comprehend why one idea was better than another.

One may think of the Consequence based design approach as the means to provide the design team with a set of rules for them to follow in the early design stage. This sounds problematic and not much different from the integrated design methods and the critique that followed (see Section 1.1). One major difference between the two approaches is that in the Consequence based design approach the rules are defined by the design team and not from outside imposing methods and criteria. The case showed that unilateral focus on energy consumption has its pitfalls, and not only on indoor environments. But when all members of the design team are included in the attempt to achieve the objectives of High-Performance Buildings and the design team's own objectives, the task of optimizing solutions becomes much more relevant for all disciplines. An energy efficient building does not in itself ensure a livable healthy and sustainable building. But the inclusion of criteria that ensures livability and sustainability changed the criteria on e.g. energy efficiency to an opportunity for the design team rather than a restriction on the design team. The design team argued that the model helps the array of disciplines participating in the early design stage (designers, architects and engineers) to focus in a more human scale with precision and priority.

Table 9 Preference towards the model features (see Section 4.1)

		Architects	Desginers	Engineers	Others	Total respondance	
Highest priority	5	4,0	5,0	4,0	-	**4,3**	**Flexibility**
	4	4,0	3,0	5,0	-	**4,0**	**Speed**
	3	2,0	4,0	2,0	-	**2,5**	**Precision**
	2	1,0	2,0	3,0	-	**1,8**	**Usability**
Lowest priority	1	4,0	1,0	1,0	-	**2,5**	**Visual Quality**

An energy renovation project can be dealt with in generic terms. "Add more insulation, new windows and put solar cells on the roof". The design team argued that the mapping method included in the model ensured that "every penny is invested in the most efficient way". This was ensured by the rating mechanism of sub categories in terms of relevance factors. Quality is however subjective, but then a subjective performance evaluation of one initiative compared to another can be very objective, if the comparison is done systematically in a transparent way. The integrated dynamic model would therefore help keep the difficult balance between quality/ subjectivity and efficiency/ objectivity. With respect to the design team's preference of the model, it is seen from Table 9 that flexibility and speed are top-ranked. Visual quality was least important for designers and engineers but architects ranked the visual perspective high. In visual terms, the model did only produce rose-diagrams with or without annotations as seen in Figure 49, and therefore its quality has much to do with how the model could be used as an illustrative tool to display solutions.

The design team ended up with a renovation solution that lived up to the Danish Low Energy Classification 2015 (Danish Building Regulations, 2013) without energy producing measures (such as solar cells). The design team argued that the model helped in making that decision.

One of the most important features of the approach and the use of the model was the ability to relate to multiple types of technical parameters and more abstract qualities. The model was in this case used to clarify both objectives and solutions between interdisciplinary focus areas. This made the model very relevant for interdisciplinary work. Energy renovation is a simple task if only defined by pure energy performance. Innovative sustainable renovation can combine more than singular well defined objectives to combine "livability" with low energy consumption in a narrow economic frame.

Identified needs for further developments

Next step for further developing the "design tool" is introducing it to the client and the residents - and improving and qualifying the areas of interest even more. The integrated dynamic model is a dialogue instrument, and it could evolve into a database for all involved stakeholders for future projects of similar character.
In terms of energy and indoor environment simulations, the present case was neither complex in terms of geometry nor in terms of the variance of solutions. For this reason, separated energy, daylight and thermal indoor environment BPS tools and models were used to inform the assessors. It would be preferred if the design team had modelled the solutions in the same design tool and the VPL could automate those assessments that were in fact quantitative.

Conclusion

The main idea to bring qualitative evaluations into the model was an important step in designing High-Performance Buildings covering all areas of sustainable thinking. The Consequence based design approach was used in this project evolving around an integrated dynamic model that is able to generate strong visual and informative feedback to the design team. In this case, the integrated dynamic model was composed of a design tool, a VPL and an assessment tool. The design team argued that quality assurance was kept by using a strict evaluation system throughout the process, and that all members of the design team were regarded as experts in their own field of discipline. The argument was that this has led to uniform and high quality assessments of the different categories and different solutions. Nonetheless, some of the decisions such as the inclusion of BMS-systems with personal monitoring were found too difficult to model with available BPS tools. Therefore, some of the energy and indoor evaluations and weights given to these decisions may have introduced bias in the system and thus in the choice of final solutions. The difficult part of qualitative assessments is to minimize such biases. From the survey the respondents agreed upon statement A, B and E (seen Section 4.1.1), hence disagreeing with the statement concerning the model's ability to improve (simulated) building performance (statement C). The reason is the lack of coupled BPS tools. The model was simply not handling simulated performance in a quantifiable way, as e.g. the model in the case study Kirk. In terms of the disagreement toward statement D, the model did not help to assist in the exploration of architectural expressions and concepts of the building design that are not directly associated with the (simulated) building performance. Again, the explanation is found in the very abstract model, which could not visualize the solutions, but only the performance of solutions.

4.1.5 Case study: Good Year

This case study goes under the pseudonym “Good Year” because of limitations and secrecy imposed by third parties. For the same reason, the real project is not displayed in any way. To document the applications of Consequence based design, the essence and concepts of the model, and where possible the collaboration between members of the design team, are therefore reproduced in another context. Some of the works found here are also found in the article “Building energy optimization in the early design stages: a simplified method” (Negendahl and Nielsen, 2015). This case study focuses on explaining the process of implementing Consequence based design and the results of this. A special focus is put on the implementation of optimization in the integrated dynamic model. Documentations consist of internal reports, interviews and surveys.

Introduction

This case study is based on an undisclosed project between the architects BIG and the consulting engineers Grontmij. The project has been put on hold for undisclosed reasons. The case is used to present the application of the method in real life design problems where architectural qualities may supersede other objectives.

From the very beginning of the design process the design team sought to avoid external solar shadings. External solar shadings were found to be expensive, difficult to maintain and difficult to incorporate in the architectural design. The design team argued that most, if not all, external solar shading systems could be avoided by carefully designing a self-shading (folded) façade (see Figure 51), however the design team had no evidence as to how this was done. Therefore, an integrated dynamic model with optimization was created to inform the design team in how such a folding façade could be designed.

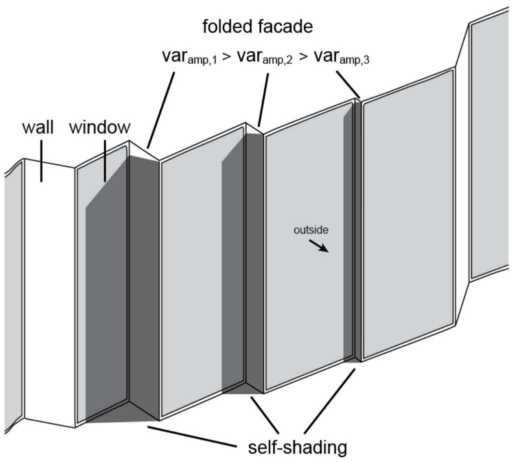


Figure 51 Folded façade concept. The amplitude of the façade folds marked by the variable $var_{amp,1\text{-}3}$ creates self-shading mechanisms on the neighboring façade unit.

By removing the external shading system as a viable design option, concerns of thermal indoor environment, building energy consumption and daylight distributions became a central part of the discussion. Four questions arose with the folded façade concept:

1. How much folding[21] is needed to avoid overheating?
2. Does increasing amplitude of folds, var_{amp} (see Figure 51) decrease the energy consumption?
3. If so, does it pay off to use more expensive high performing glazing types[22]?
4. How does the folding affect the daylight distributions in the offices?

To answer these questions, it was decided to make use of a multivariate optimization method to explore the many solutions where folding could influence the energy consumption, the daylight distributions and indoor thermal environment while considering the cost of the window systems.
It was decided among the members of the design team to use an entire building evaluation process of energy, cost, daylight and thermal indoor environment; this was decided because the building designers wanted to control a continuous and changing façade around the building.

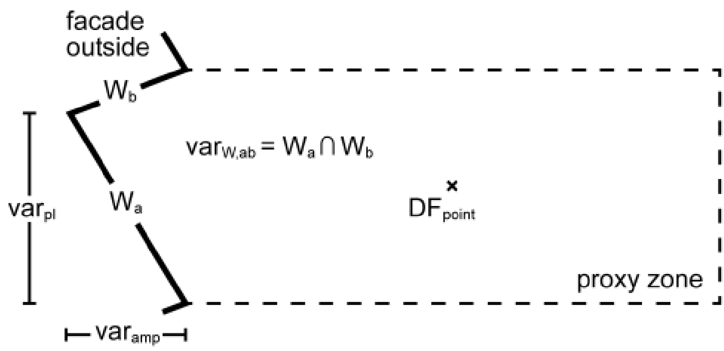


Figure 52 Plan view of a proxy zone represented as a dashed line. The variables are used to constrain the optimization process

The use of simple representatives of rooms (e.g. by simulating variations of rooms) was found by the simulationists in the team to be unfit for this process as the small and continually connected variations on the façade would create too many possible combinations and thus too many simulations. What was needed was a very fast entire building simulation that could (to an acceptable level of precision) present the entire building energy consumption, the price of the façade, the amount of daylight in every room and estimate the risk of thermal overheating

[21] Amount of folding is determined by adjusting amplitude var_{amp} and window size var_{pl} (see Figure 52) and var_{blend} (see Figure 53)

[22] High performing glazing types: window panes with reduced convection and radiation heat losses (low U-values) and/ or reduced solar heat gain coefficients (low g-values)

problems inside the building. To do this, the building needed to be divided into thermal zones and simulating each zone would be necessary. However, at this point in the design process room placements were not fixed; consequently, any zone division was very dubious and would affect the simulations significantly. It was for this reason decided to use proxy zones instead of actual room geometry. The proxy zone as seen in Figure 52 is defined by a volume extruded into the building in a fixed depth (here 5m) from the façade where the window W_a or W_b is positioned.

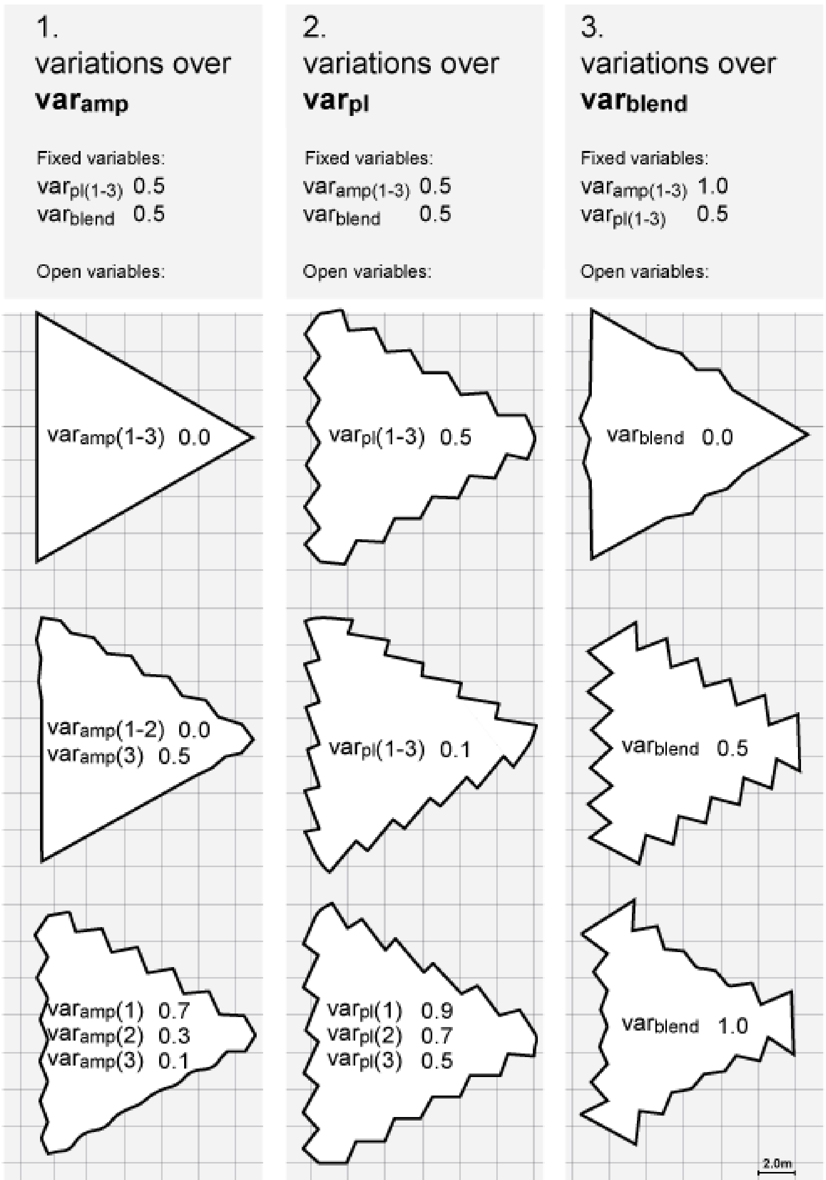


Figure 53 Plan views of a small building example to explain the changes in design variables. 1. shows variations over var_{amp}. 2. shows variations over var_{pl}. 3. shows variations over var_{blend}

As mentioned on the previous page, the building designers valued a continuous façade, where one fold were mostly similar to the neighboring folds; thereby, only subtle changes from one façade fold to the next was allowed. In terms of optimization, this is a complex type of dynamic constraint. However, the implementation of this type of constraint functions is straightforward when VPLs are present in the model environment. The design team's solution is a scripted function that utilizes the hyper parameters $var_{amp(1\text{-}3)}$, $var_{pl(1\text{-}3)}$, var_{blend} to control the folding. $var_{amp(1\text{-}3)}$ controls the amplitude in on the three façades while $var_{pl(1\text{-}3)}$ controls the vertical placement of the fold on each façade and var_{blend} adjusts the "blending effect" that intermixes the folding between façades.
Figure 53 shows variations of the hyper parameters; for example, var_{pl} shifts the fold clockwise with small values and $var_{amp(3)}$ controls the north eastern façade.
By defining these geometrical constraints, the idea was to explore the many different "optimal" solutions that were provided from the optimization process. The different solutions shown in
Figure 53 do not represent any architectural preferred strategy, but show the impact of the design variables.

The integrated dynamic model

A large effort was put into the choice of simulation tools. Table 10 and Figure 54 describes three different BPS tools applied in the method; all chosen by the simulationists for their ability to evaluate performance with minimum computational power and dynamically deliver the results back into the model.

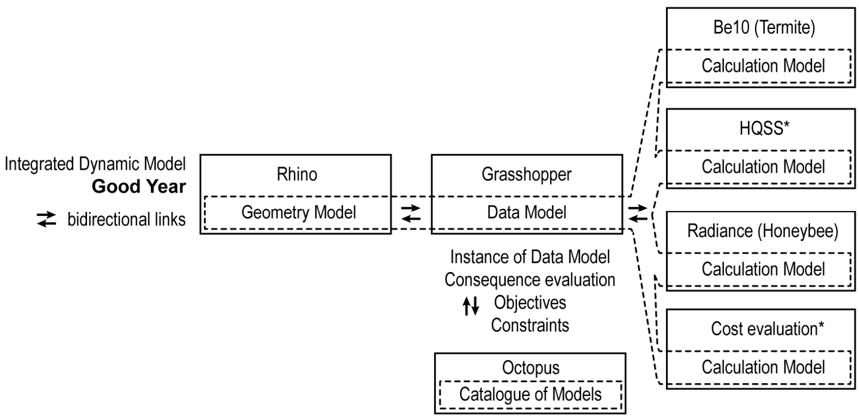


Figure 54 The integrated dynamic model. The design tool Rhino combined with the VPL Grasshopper and the BPSs Be10 (Termite), HQSS*, Radiance (Honneybee) and Cost evaluation*. *created in Python and Grasshopper

Radiance (Ward et al., 1988) (Evaluation of daylight, Table 10) is processed through the interface Honeybee (Roudsari et al., 2014) while Be10 (SBI, 2013) (Evaluation of building energy consumption, Table 10) is processed through the interface Termite (Negendahl, 2014c). Be10/Termite performs monthly averaged quasi-steady-state calculations and is used in Denmark to evaluate energy consumption of all new buildings. The hourly quasi-steady-state method (shortened HQSS) (Evaluation of thermal overheating, Table 10) is written in Python and Grasshopper and is based on ISO 13790 (ISO, 2008).

The monthly calculation performed by Be10/Termite gives accurate results on an annual basis as demonstrated by Christensen et al. (Christensen et al., 2013). But the results for individual months close to the beginning and the end of the heating and cooling season can have large relative errors (ISO, 2008). Monthly quasi-steady-state calculations may be sufficient to estimate building energy use but they are considered too uncertain as a method to estimate thermal indoor environment. For this reason an alternative quasi-steady-state method for hourly calculations has been added to the model.

The HQSS tool facilitates the calculation using hourly user schedules (such as temperature set-points, ventilation modes and hourly control options based on outdoor or indoor climatic conditions). The tool produces hourly results, but similar to other quasi-steady-state hourly calculation methods, the results for individual hours are not validated and individual hourly values can have large relative errors (ISO, 2008). Nevertheless, for early design stage estimations the use of hourly calculation methods is expected sufficient in detail and precision. The HQSS tool is used to estimate an average hourly heat balance to determine whether the cooling load can sustain the internal and external heat gains.

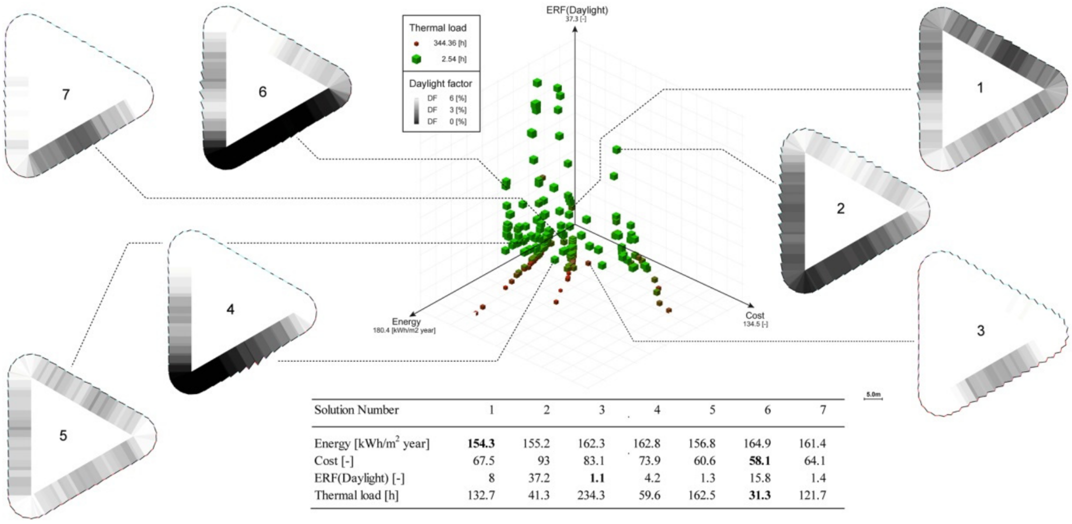
The cost evaluation was simply defined as the cost of windows. This was based on window type and areas of windows.

Worth noting is that the computing power of using hourly calculation is around two orders of magnitude more intensive than divisional period (e.g. monthly) quasi-steady-state methods. However, this is still at least one order of magnitude less computationally intensive than detailed dynamic simulation methods.

Table 10 BPS tools applied to the method

	Objective	Tool	Implementation
1	Evaluation of daylight	Radiance [11]	Honeybee [30]
2	Evaluation of building energy consumption	Be10 [12]	Termite [31]
3	Evaluation of thermal overheating	HQSS	new Hourly Quasi-Steady-State implementation

The multivariate optimization procedure was performed at dual-core laptop over a period of three days. A population size of 300 ran through 32 generations of SPEA2 (Zitzler et al., 2001) trials with the optimization tool Octopus (Vierlinger, 2014), which turned out to be sufficient for convergence. On average, every simulation/evaluation of the four criteria took less than 30 seconds. This is considered very fast when dealing with entire building simulations on regular PCs.



Solution Number	1	2	3	4	5	6	7
Energy [kWh/m^2 year]	**154.3**	155.2	162.3	162.8	156.8	164.9	161.4
Cost [-]	67.5	93	83.1	73.9	60.6	**58.1**	64.1
ERF(Daylight) [-]	8	37.2	**1.1**	4.2	1.3	15.8	1.4
Thermal load [h]	132.7	41.3	234.3	59.6	162.5	**31.3**	121.7

Figure 55 The plot in the middle shows axes of a Cartesian space (x, y, z), where x is energy [kWh/m^2 year], y is Cost [-] based on the cost function of windows and z is ERF(Daylight), which represents the penalized function of Daylight factors ERF(DF%). The box size and color describe the amount of hours [h] above the maximum cooling capacity. The plan view of 7 selected solutions is shown in the solution space, daylight factors in each zone are plotted as a grey scale hatch. The table in the bottom shows details on the objectives for the selected solutions

In this case seven selected solutions are presented (Figure 55), and as stated previously this case is synthetic, the actual choice of solutions is of minor interest. Therefore, focus is put on the process to define and create the integrated dynamic models, not the results from the models.

It is up to the design team to choose which overall tradeoff strategy suits the design better. The seven choices of solutions show that a very diverse façade composition with a large amount of folds may be optimal if daylight and thermal environment is valued high, but in terms of capital-cost and annual energy costs a uniform and almost flat façade composition is better performing.

Results

This case considers a wide range of problems when BPS tools are used to optimize buildings in the early design stages. One is the actual use of optimization methods in early design stages, which clearly has its limitations, as machine automation is very difficult to combine with quality-defined objectives. Souza & Knight 2007 warned that the distance between those that simulate and those that design may be one of the largest problems when using optimization methods in early design stages: *Setting up criteria to evaluate performance and relate these criteria directly to design actions is a methodological problem independent of the simulation tool being used. It requires simulationists to fully understand the way*

designers think, i.e. essentially exploring interactions of all parameters together and dealing with all the variables at the same time. (Souza and Knight, 2007)
To a great extent, this can be solved by utilizing an integrated dynamic model where both the simulationists and the designers work in a fully coupled environment (Negendahl, 2014b). In this case, the design team that consists of building designers and simulationists was able to develop an integrated dynamic model that took both qualitative and performance based criteria into account. This evidently leverages some of the quality assurances mentioned by Hensen (Hensen, 2004), such as using appropriate levels of model resolution for the early design stage and requirements for sufficient domain knowledge by the users.

Table 11 Level of agreement to the five statements A-E (see Section 4.1)

		Architects	Desginers	Engineers	Others	Total respondance	
Agree completely	100	-	100	75	60.0	**78**	**Statement A)**
Agree very much	80	.	70	60	60.0	**63**	**Statement B)**
Agree mostly	60	75	50	100	75.0	**75**	**Statement C)**
Disagree mostly	40	50	0	50	75.0	**44**	**Statement D)**
Disagree completely	0	50	100	100	90.0	**85**	**Statement E)**

The model itself was part of the design process that contributed in the decision making of how to design the façade. Therefore, the design team did not consider the optimization method as a definite form finding process but more a as means to extract valuable information from an open ended design problem. This is seen in Table 11, Statement D where most designers and architects either disagreed or did not see that the model assisted in the exploration of architectural expressions and concepts of the building design that are not directly associated with the (simulated) building performance. But it had positively assisted in achieving High-Performing buildings without compromising the architectural expressions and concepts of the building design (Statement E). In an interview with one of the architects, he declared that he was not aware that a "full-blown" optimization process had been used to inform the design, only that the suggested solutions presented by the simulationists were the results of a typical energy simulation. This suggests that the integrated dynamic models using optimization are regarded as any other integrated dynamic model.

In relation to facilitation speed and the method's ability to *shorten synthesis analysis evaluation cycles*, as noted by Mora et al. (2008) and Struck et al. (2009), the integrated dynamic model was able to generate a new result in less than 30 seconds on a fairly modest two-core laptop. Speed and flexibility were among the top-ranked features of the model according to the design team (Table 12).

Table 12 Preference towards the model features (see Section 4.1)

		Architects	Desginers	Engineers	Others	Total respondance	
Highest priority	5	5,0	5,0	2,0	3,0	3,8	Flexibility
	4	4,0	3,0	4,0	4,0	3,8	Speed
	3	1,0	4,0	5,0	1,0	2,8	Precision
	2	2,0	2,0	1,0	2,0	1,8	Usability
Lowest priority	1	3,0	1,0	3,0	5,0	3,0	Visual Quality

The flexibility of the integrated dynamic model meant that the objectives and constraints of the optimization could be adjusted to fit the design process and not the other way around. Even though much of the process of generating solutions was part of automation processes, the actual value of the method is found in the consequence feedback. In other words, the value is found in the facilitation of the design rather than in the automation of the design.

The BPS tools used by the model are integrated and fast, but it comes with a cost of validity and precision. The annual energy simulations based on Be10 merely present a trend in energy consumption when the geometry in the model is changed in marginal steps. For this reason, small façade changes will not affect the energy use significantly. The dynamic effects of the building, use e.g. drawing curtains when the sun creates glaring effects in offices, are not taken into account. And many similar dynamic effects, which are not considered, may result in inaccurate daylight and energy evaluations. More on this discussion is found in (Negendahl and Nielsen, 2015). In terms of the design team's preference of precision, the designers and the engineers ranked precision higher than the architects and others (Table 12). Therefore, in similar future projects more precise simulations may be needed to improve the feedback quality.

Conclusions

Multivariate optimization combined with simplified building performance tools demonstrated the finding of Pareto optimal solutions in reasonable computational time. It is clear that an integration of optimization algorithms can drastically change the time consumption of performance analyses within architectural design processes, allowing designers to focus their attention on taking informed design decisions. It is concluded that quasi-steady-state methods implemented as part of integrated dynamic models are fast and flexible enough to support building energy consumption, indoor environment and cost optimization the early design stages. Additionally, these types of models showed potential to integrate various types of architectural constraints in the optimization process, thereby integrating the domains of the building designer and the simulationist through a common platform.

When it comes to using the combined evaluations with stochastic optimization algorithms (like the SPEA2 algorithm demonstrated), it can be concluded that the level of precision is sufficient for the initial design approach, but more precise evaluation methods are needed in later stages when more detailed design options have been settled.

It is concluded that the respondents agreed upon statement A,B,C and E (see Section 4.1.1) The model was not found to assist in the exploration of architectural expressions and concepts of the building design that are not directly associated with the (simulated) building performance (statement D).

4.2 Concluding remarks of the case studies

To briefly collect the findings of the five cases this section explains how the findings correlate with the approach of Consequence based design with the singular focus on tools.
First, the various integrated dynamic models are discussed in terms of:

- Flexibility (of features and complexity)
- Speed (of performance feedback)
- Precision (of the BPS)
- Usability (i.e. easy to use)
- Visual quality (for presentation)

Second, the use of the Consequence based design approach is discussed in terms of how the design teams used the integrated dynamic model. To structure this discussion and to compare the different case studies, the integrated dynamic model is considered as a tool to enable certain objectives. Hence, the integrated dynamic models are viewed as the ability to support the design team as a tool:

- Sketching tool
- Communication tool
- Calculation tool
- Collaboration tool

Based on the case studies the approach of Consequence based design is discussed in relation to the research aims stated in 1.4, the scientific goals in section 1.5 and development goals in section 1.6. This discussion includes how widely applicable integrated dynamic models are and what circumstances are needed to apply the Consequence based design approach for new projects.

4.2.1 Flexibility

The flexibility of the different integrated dynamic models was directly coupled to the amount of anticipated changes and the amount of open design variables, and it is coupled to the type of performance analysis required from the model. The amount of analytical complexity over the parametric flexibility of the involved projects is visualized in Figure 56. The analytical complexity is categorized as the involvement of multi-disciplinary analyses, the depth of those analyses and the flexibility of the coupled BPS tools. The parametric flexibility is primarily associated with geometric parametrical flexibility, but also the amount of open variables (levels of freedom) relating to the BPS tools influences the parametric flexibility.

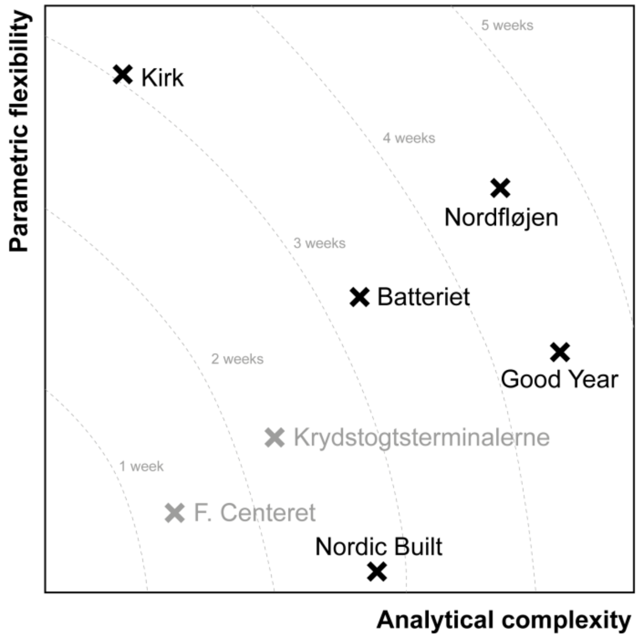


Figure 56 The flexibility in terms of parametric flexibility and analytical complexity of the involved projects. In grey the number of weeks (week = 37 hours) one person needs to define, develop and implement an integrated dynamic model.

The case of Kirk is the most flexible project in the parametric sense. The sheer amount of open variables modelled for this project would have been very difficult to manually reproduce if the BPS tool and the design tool had not been coupled. Based on this alone, if this project was approached in a more traditional fashion the geometry model had to be much simpler for the purpose of analyzing the energy performance. As the building had multiple double curved surfaces intersecting, it was difficult for the design tool Revit to handle it. The integrated dynamic model did not show any similar problems in representing the geometry, which increased the flexibility of the model. However, the high level of flexibility increased the risk of errors in the integrated dynamic model. The simulationists concluded that to withstand the increasing number of errors, the model had to include some sort of automated quality checks. For example aggregated areas could be automatically calculated and their numeric values tested within certain bounds, similar quality checks could be placed on the output from the individual BPS tools.

The Good Year project on the scale of analytical complexity is the most challenging of the involved projects. This was due to the complexity of the multivariate optimization and the need for development of new BPS tools (HQSS). The model was rather limited in terms of flexibility of features, but the number of design variables and their highly irregular effects on the façade enabled the model to tackle almost any “folding façade” solution. This high level of parametric variation in the model, and the amount of BPS tools coupled to the model made the model more flexible. Nordic Built and Batteriet had almost equal magnitude of analytical complexity in the model, but Batteriet was more challenging in the sense that the model was coupled to actual BPS tools whereas the Nordic Built project model had very little parametric flexibility as all analysis was quality defined or based on separate tools. Nordfløjen was the most flexible of the projects that also had a high level of analytical complexity. The flexibility increased as the model was in fact distributed among different design tools and VPLs and used two BPS tools to generate consequence feedbacks.

From Figure 56 it is possible to estimate the number of worker hours needed to implement new integrated dynamic models in new projects. What is needed is an expectation of the parametric flexibility and analytical complexity of the project. It would be impossible to forecast every single requirement of such a model. However, based on the experience documented here, the level of complexities and the necessary flexibility are something that can be controlled by the design team. The building designers may choose to model highly complex geometry and define flexible layouts for buildings, as long as they are aware that this process takes more time to analyze. The same can be said for the simulationists; as long as they know the time is available, they can “open new variables” and new performance evaluations in the model. In terms of what type of analyses is needed for a project, no general rule applies. In-depth analysis might be required in one certain aspect of one project, but not in another, therefore it is necessary to carefully choose the BPS tool, the analysis type and the number open variables (as discussed in Negendahl (Negendahl and Nielsen, 2015)). One lesson learned from Nordic Built is that when qualitative assessments are taken into account and defined as equal as the quantitative performance evaluations, the design team is better at focusing on the objectives of High-Performance Buildings. Therefore, integrated dynamic models need to be better able to include qualitative evaluations while maintaining a high flexibility.

In general, more complex projects are more vulnerable to changes that are not included in the integrated dynamic model. This could be changes in the building program or other unexpected changes imposed from external sources. And similar to Harding et al.’s (2012) experience, *hacking away inefficiently at the parametric topology* is the only option available to fit the model into new constrictions.

4.2.2 Speed

The fastest model in terms of analytical feedback is the model from the Nordic Built case. This is simply because no BPS tool was coupled, and thus no “waiting time” for the solver to finish its simulation was present. The fastest method that includes an actual performance evaluation is the case of Batteriet. In general, an evaluation took between 1 and 2 seconds, and the evaluation time was mainly due to limitations of the graph-based modeling techniques. An implementation in a lower level programming language such as Python is considered to improve the speed. The Nordfløjen project used Ecotect as a BPS to estimate daylight potentials; this simulation took around 30 minutes, however this simulation was only performed when the geometrical base model was changed. The simulation with Be10 took 1-2 minutes, a rather slow process that was caused by a complex parametric model, again caused by the limitations of the large graph-based VPL. This was even more apparent in the Kirk case. The model took around 5-10 minutes to simulate the energy consumption of the building, therefore the complexity scales with the feedback speed. Good Year was the only project of the five that utilized optimization algorithms. Here each simulation took around 30 seconds (this includes an entire building energy analysis, daylight analysis in every zone, cost-analysis and evaluation of overheating hours in every zone); this is considered extremely fast compared to the speed of existing BPS tools (Negendahl and Nielsen, 2015). However, as optimization is used the entire modeling sequence took around three days.

4.2.3 Precision

In terms of the need for high quality assurance the models coupled to existing validated BPS tools are more reliable and robust (Hensen, 2002). Therefore, the projects Nordfløjen, Kirk and Good Year are regarded as more reliable in terms of energy performance related feedbacks compared with Batteriet and Nordic Built. However, the quality assurance is not ensured just by coupling the right tools to the model, it matters how the tools are used. As stated in (Negendahl, 2015a) there are many factors that contribute to the quality assurances, therefore it is difficult to compare the different projects in terms of validity. What may be concluded is that compared to traditional approaches (e.g. as mentioned in Section 2.3) the Consequence based design approach has ensured much higher quality assurance for the simulationists in the design team, simply because simulationists are active members of the design team in the early design stage.

4.2.4 Usability

All cases used at least one integrated dynamic model based on Rhino as a design tool and Grasshopper as VPL. Only the BPS tools separated the different cases. Usability is simply interpreted as a matter of the usability of the coupled tools and how capable the members of the design team were in handling the mentioned tools. In all cases the simulationists were the main developers and operators of the models, therefore the usability in the five case studies has been most important for the simulationists. What can be said in general on usability on integrated dynamic models is that Grasshopper has improved significantly from when the Ph.D. project began back in December 2011. Many new features have been added and more developers have added plugins and to the VPL, which makes many complicated automations much easier to implement today, than just a few years back.

4.2.5 Visual quality

The visual quality between the case studies ranged from simple representations of windows (e.g. Batteriet) to very complex double curved surfaces in the Kirk projekt. The visual quality was not the primary objective in any of the projects, but as it was seen, members of the design team valued the visual quality very high for some projects, while in other projects it was found of less importance.

4.2.6 Sketching tool

None of the models have been considered as being equivalent to a sketching tool. One could imagine that the parametric capabilities of the VPL in the combination with the design tool could give rise to fast sketch like tool properties. This does not seem to be the case. In all of the case studies the integrated dynamic models were used to inform the design in a more clearly (objective) defined design space, less open for drastic changes often associated with sketching. Batteriet, Nordfløjen, Kirk and Good Year were all concepts developed over previous design iterations. Only the façades were considered truly open for changes. The Nordic Built model could be considered as a sketching tool for criteria and project formulation, but not a sketching tool for building design.

4.2.7 Communication tool

The Nordic Built case was the only case that directly argued that the model acted as a communication tool. However, all cases to a certain extent had used the model to communicate the results of the process to either the design team itself or to other stakeholders.

4.2.8 Calculation tool

It is important to differentiate the processes of automations. Generative mechanisms or ontology based rules are not the same as when the model is *calculating.* Calculating is here the process of what usually defines a typical feed forward approach, see Section 2.4. Batteriet, Nordfløjen and Good Year have used one or several feed forward methods. Where Kirk and Nordic Built were either completely dependent on external couplings of an existing BPS tools, or by human assessments, Batteriet, Nordfløjen and Good Year used run-time speed methods to evaluate either energy performance, indoor environment performance or both. There are endless possibilities in how the model can act as a calculation tool. What is important to repeat is that quality assurances need to follow the calculation methods. In the case studies that used monthly quasi-steady-state methods (Termite/Be10) it was argued that such method is sufficient in terms of precision for the early design stage. However, in later stages, when more information about the building has been established it is necessary to use more advanced and precise methods to verify the design decisions. In general it must be concluded that the calculation methods have been found valid for the early design stage.

4.2.9 Collaboration tool

The integrated dynamic models have been approached in terms of different levels of collaboration. In other words, in some projects e.g. Kirk in Section 0 the utilized model had been primarily accessed by the simulationists (consulting engineers) in the design team. And in other cases e.g. Nordfløjen in Section 0 and Nordic Built in Section 4.1.4, both the building designers and the simulationists cooperated in the definition of the scope and the implementation of the model. It was found to be difficult to compare the effect of the increased integration of disciplines in the models, hence to evaluate in which strategy it is better. But in general terms, it can be concluded that the amount of involvement of all disciplines in the process defining the criteria for the integrated dynamic model is important if the building design needs to include all aspects of a design. If the simulationists took the task of defining the model criteria alone, the focus would be to solve the objectives of High-Performance Buildings this would likely increase the chance to create High-Performance Buildings. Such a model, with clearly defined criteria, does not necessarily need parametric analyses on detailed geometry or do not need to consider design aspects beyond the scope of High-Performance Buildings. However there is no guarantee that the rest of the design team will use the model, if the rest of the design team's objectives have not been considered. And therefore, there is a need to continuously pursue the integration of multidisciplinary performance evaluations in the model as well as the inclusion of multidisciplinary operation of distributed models.

4.3 Practitioner perspectives

In the aim to generate a wider assessment on integrated dynamic models and the approach of Consequence based design surveys have been sent to practitioners in the field of parametric design and engineering. The surveys have been posted on the open user forums of the VPLs Grasshopper and Dynamo and sent directly via email to certain practitioners, who the author had interviewed informally in relation to other projects and to external stays in the US and the UK. 16 respondents gave feedback relevant to the survey; they can be found in Appendix B. None of these respondents have been associated with any of the five case studies. The number of feedbacks is not large enough to establish a statistical conclusion. However, the number of practitioners worldwide that frequently use integrated dynamic models is (still) small, as the method is quite new, undeveloped and unknown. Therefore, the conclusions of this section must be seen as a trend.

4.3.1 Survey results

In relation to the five statements found in Section 4.1.1, there seems to be a general agreement among practitioners in the field (see Table 13) that integrated dynamic models are able to:

- **A)** Create a better common starting point of collaboration between the members in the design team
- **B)** Improve the communication between the members in the design team
- **C)** Improve the (simulated) performance of the building design
- **E)** Positively assist in achieving high performing buildings without compromising the architectural expressions and concepts of the building design

Only statement **D)** regarding how the model can assist in the exploration of architectural expressions and concepts of the building design that are not directly associated with the (simulated) building performance, there was a slight disagreement among non-simulationists, and non-building designers. This corresponds well with the survey results from the five cases studies.

Table 13 Level of agreement to the five statements A-E (see Section 4.1)

	Scale	Architects	Desginers	Engineers	Others	Total respondance	
Agree completely	100	80	-	92	60	**81**	**Statement A)**
Agree very much	80	79	-	100	40	**82**	**Statement B)**
Agree mostly	60	100	88	83	70	**87**	**Statement C)**
Disagree mostly	40	68	82	79	48	**69**	**Statement D)**
Disagree completely	0	80	77	93	50	**79**	**Statement E)**

In terms of preference towards model features, the practitioners in general ranked flexibility and speed highest, see also Table 14. Precision and usability were ranked as equally important and visual quality as least important. In general, architects and engineers ranked flexibility highest, but what could come as a surprise is that engineers ranked precision as least important where architects valued precision higher. Also engineers valued visual quality much higher than architects. These two differences could be coupled with the use of the model as a communication tool and a calculation tool. The typical engineer acts as a simulationist and the typical architect acts as a building designer. The simulationist will use the model as a flexible tool to display building performance consequences to the building designer. When building designers use the models, they are less dependent on simulationists, and therefore the quality assurance associated with precision is valued higher than visual quality.

Table 14 Preference towards the model features (see Section 4.1)

	Scale	Architects	Desginers	Engineers	Others	Total respondance	
Highest priority	5	4,1	2,0	4,2	3,0	**3,9**	**Flexibility**
	4	3,3	5,0	3,2	3,0	**3,3**	**Speed**
	3	2,9	3,0	2,2	3,0	**2,7**	**Precision**
	2	3,1	1,0	2,4	3,0	**2,7**	**Usability**
Lowest priority	1	1,6	4,0	3,0	3,0	**2,4**	**Visual Quality**

In the question on the importance between speed and precision, no consensus can be seen. In Table 14, model speed seems to be more important, but when it comes to the option between the two, 40% prefer speed over precision and 40% prefer the opposite, while only 20% rank speed and precision equally (see Figure 57). This means that speed and precision are project specific, and it matters much that the practitioners have the option to shift between high speed (low precision) and high precision (low speed). Almost the same can be concluded with the choice between speed and visual quality, but most practitioners prefer speed over the visual quality. When it comes to flexibility over usability there is a clear preference towards flexibility.

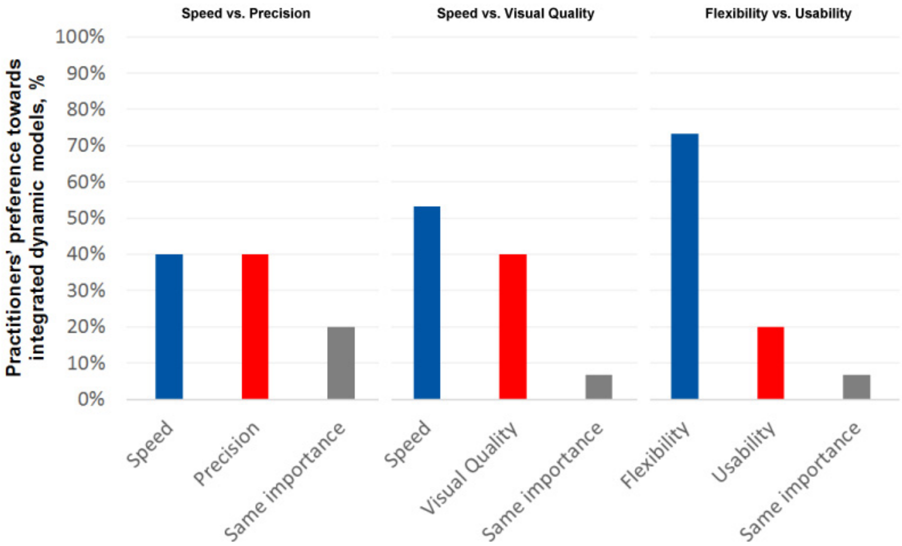


Figure 57 Practitioners preferences (% surveyed): Speed vs Precision, Speed vs Visual Quality, Flexibility vs Usability

A few of the respondents pointed to the value of the model in being distributed between tools. For example, one practitioner stressed that independencies between models assured higher levels of coordination in the design team. Another mentioned that having analytical models downstream from geometric/generative models enables the design team to discuss design options. This make the team consider (more) options and broaden design decisions to external parties.

In the question on how integrated dynamic models were used as a tool in the early design stage, four options were available: Sketching tool, Communication tool, Calculation tool and Collaboration tool.
In the relative comparison of the four tool types most practitioners agreed with the model comparison to a calculation tool (see Figure 58). Most practitioners disagreed with the comparison to a communication tool, (even though 60% still used the model as a communication tool). Around 69% of the practitioners used the model as a sketching tool and 76% of the practitioners used the model as a collaboration tool. In terms of how practitioners want to use integrated dynamic models in the future, most practitioners liked to use the model as a calculation tool as seen in Figure 59. Least interest was found in the tool comparison of sketching and communication capabilities. Collaboration capabilities were ranked slightly higher. Nevertheless, there is high interest in improving all four tool

capabilities for future developments. In relation to the future developments some practitioners noted that the limitation of the model is coupled with the limitations of the BPS tools, and more features and flexibility would increase the feedback process to the designer.

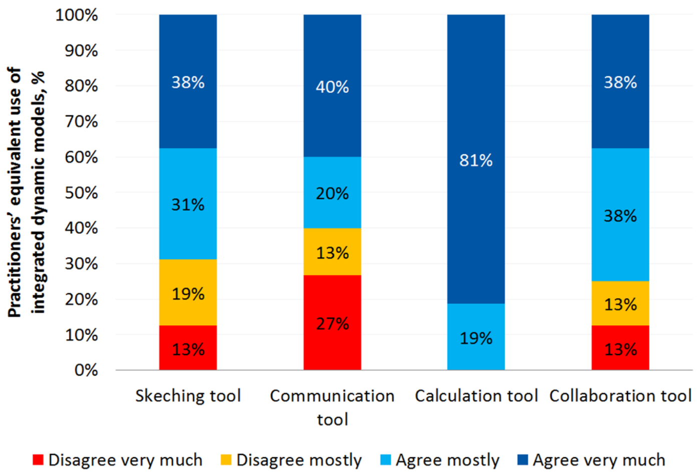


Figure 58 Practitioners' general use of integrated dynamic model (as used today)

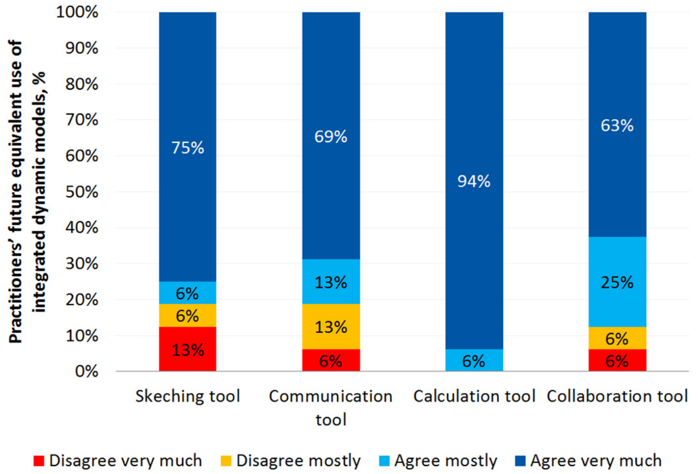


Figure 59 Practitioners' general use of integrated dynamic model (as liked to be used in future projects)

To the question whether integrated dynamic models have positively contributed in improving building energy performance, indoor environment and/or sustainability all[23] practitioners except one answered: *yes*. Some of the practitioners noted that improving performance was not necessarily the main

[23] Two practitioners answered the question in relation to structural performance

objective, but the model still helped the designer to improve the performance. One practitioner mentioned that optimization can be an important part of the model, especially when used with evolutionary algorithms to find best results among conflicting design criteria. Also the use of integrated dynamic models *helps in real-time feedback to improve the design proposal*. Of course this only works if the coupled BPS tool delivers real time feedback. Several of the practitioners have used the models on façade design, which includes analysis of daylight and energy. The one practitioner who answered *no* to the question used the model as a sketch without the concern of performance, as this practitioner was interested in the resulting shape. This way of using integrated dynamic models can raise concern, as the actual building performance is of little interest to the designer. Nevertheless, it is fascinating that even if the designer has more focus on generating shape, the integrated dynamic model can be a means to find such a shape. And if the sufficient quality assurance is addressed, the resulting shape may very well improve the performance of any objective, even though the designer has no intention to do so.

4.3.2 Conclusion

In general, integrated dynamic models create better common starting points of collaboration between members in the design teams among practitioners. The models improve the communication between the members in the design team, and in every case, where it is the intention of improving High-Performance Building design, the design has been improved. In addition to these improvements the integrated dynamic models made it possible to sustain the architectural expressions and concepts of the building designs as these elements were a central part of the model. Practitioners in general prefer models with a high level of flexibility. This includes features of all the coupled tools, and the option to include e.g. optimization algorithms into the model. The tradeoff between speed and precision depends on the specific case. Therefore, there is a need for very fast BPS tools and very precise BPS tools. Most practitioners valued high levels of collaboration within the model, which includes a wide range of disciplinary expertise. The practitioners' perspectives are highly in line with the findings in the five case studies.

4.4 Developments - to support Consequence based design

The most central needs for developments have been identified from the case study results in sections 4.2.1 to 4.2.9. Along with the practitioners' perspectives seen in Section 4.3, these needs can be categorized as four main challenges with the Consequence based design approach and the integrated dynamic models:

- The integrated dynamic models are limited by slow consequence feedback
- The integrated dynamic models are challenged by low quality assurances
- The integrated dynamic models need better tools to qualify subjective assessments
- The integrated dynamic models need better methods to combine qualitative and quantitative analyses when optimization is to be used

As seen in Figure 29 the "Developments" have been placed on a timeline along with the case studies. The developments have been directly coupled to the needs / challenges identified above.

In the following sections, each development is introduced in relation to specific needs identified in one or several case studies. Next, each development is briefly discussed and in the attempt to solve one or more of these challenges, the developments conclude how integrated dynamic models and/or Consequence based design benefit from these developments.

4.4.1 Development: Decision diagrams

A lesson learned from several of the projects, was that communicating decisions within a design team is very challenging. This was especially the case when the integrated dynamic model had to visualize various non-geometrical design variables or the effect of more indirect performances that were due to whole building evaluations. A systematic way to communicate and display the consequence of any performance parameter of any design variable is needed.

Introduction

To make the design exploration computationally feasible, Hopfe and Hensen (2011) argued that the analysis of sensitive variables is a good starting point for a more integrated design analysis. The problem with open ended design problems such as building design is that the sensitivity analysis from one project is not necessarily applicable to the next project. The analysis must be project specific. One way to determine which variables are more sensitive is to utilize Monte Carlo methods, e.g. (Burhenne et al., 2011; Hopfe and Hensen, 2011) or stochastic analyses, e.g. (Eisenhower et al., 2012), in coupled tools, which is possible with integrated dynamic models (Negendahl and Nielsen, 2015). However, for a very high number of variables that affect the solution to a great extent (e.g. number of floors) the methods are time consuming and difficult to use in practice (Hopfe, 2009). Therefore, a much more simple and direct way to identify the most sensitive variables is needed.

One example is the need to get a clear overview of how to achieve a DGNB Gold certification or to fulfill the Danish energy and indoor environment regulations. The proposed decision diagram is a simple way to display the consequences of sensitivity analyses. The method is based on project specific building design and it enables the design team to navigate securely towards (predefined) goals and assure that the choices consider the multiple factors involved. The current examples were extracted from the Good Year case and anonymized for the same reasons as stated in Section 4.1.

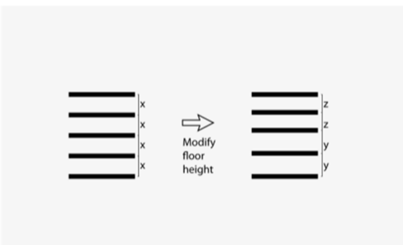


Figure 60 changing floor heights. Distributing floor heights according to floor level can increase daylight in the lower floors where daylight is sparse.

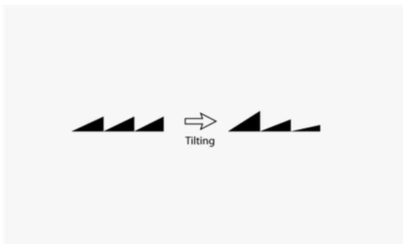


Figure 61 Tilting/angulation of separate façade sections. On the smallest scale, the individual façade sections can be oriented to either reduce or improve the amount of direct sunlight.

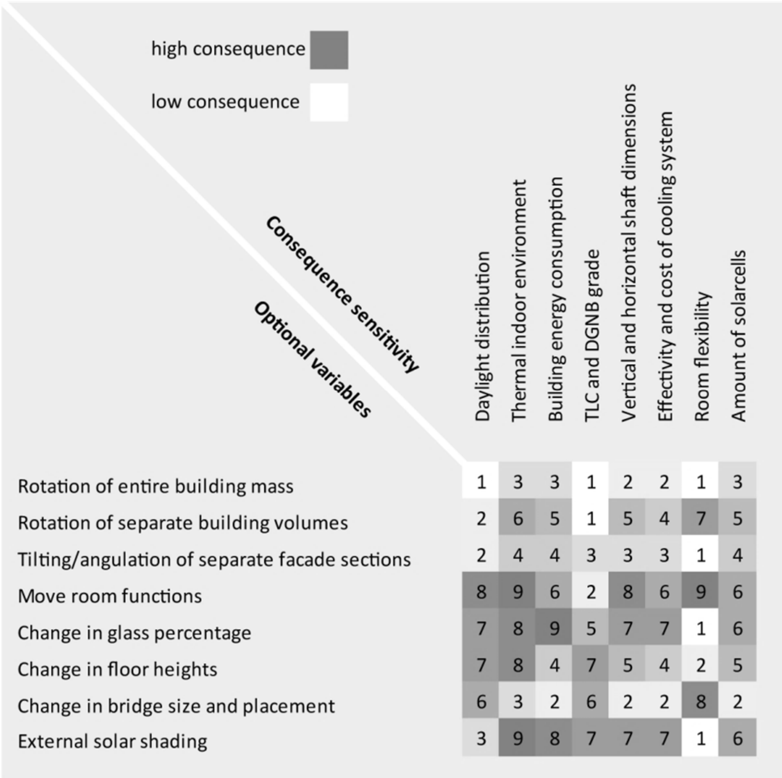


Figure 62 Example of a decision diagram. It shows the consequence of changes in terms of performance, e.g. energy performance. Here it is apparent that moving some room functions and changing the glass percentage has the largest influence on the energy and sustainability parameters. Image by courtesy of Grontmij ©

The decision diagram in Figure 62 illustrates the optional (design) variables (also illustrated in Figure 60 and 61) and their consequences in terms of energy performance. The chart clarifies and simplifies the complexity within each choice by defining how all variables depend on one another. In this particular case the cumulative or discharging effect of each variable is not accounted for. In this particular example the numbers are simply generated, but they may have come from in depth analyses or previous experiences. Ideally, the design variables will be handled in an integrated dynamic model along with precise evaluations of the topical performance. The chart may be used to identify the starting point for design optimization. Or it may be used to identify solutions that are less complex to model to begin with. In this regard the chart may assist in planning an integrated dynamic model. It can clarify which type of variables must be modeled more precisely and in which case the design team may estimate performances. The chart may also help (much like the model in Nordic Built) to communicate certain key preferences within the design team.

Conclusion

The decision diagram may be visualized as a matrix or a rose-diagram (like seen in the Nordic Built project in Section 4.1.4). The important thing to notice is that visualizing the *consequence* of particular design variables may change the design direction completely. Therefore, the simulationists who create such visualization in form of charts, matrices or diagrams must take extra care in identifying the most *important* variables. The most important does not necessarily mean the most sensitive design variables, at least not to all members of the design team. Therefore, to better visualize the importance of building performance the simulationist needs to take the building designer's concepts into consideration. Decision diagrams can help the design team in the pre-modeling stage to identify the most important way to define the open design variables, to efficiently structure integrated design modeling. This pre-process should be included in any Consequence based design approach to reduce the time spent on modeling parametric relationships and choose which analyses to focus on.

4.4.2 Development: AHP

The idea to utilize an Analytic Hierarchy Process (AHP) came out of the quality assurance challenges relating to the qualitative assessment methods that have been used in e.g. the case of Nordic Built. Saaty (1977) created AHP in the late 1970s in order to determine the relative importance of each variable in the decision making matrix on a pair-wise basis. The method deals with independences among variables or clusters of decision structure to combine the statistic and judgmental information, and it has been used in multiple research disciplines (Uzoka et al., 2011). Using AHP is an alternative to the various Multi Criteria Decision Methods suggested for use within integrated design, for example MCDM-23 (Balcomb et al., 2002). The advantage of the AHP is that it offers a formal and logical way of including qualitative values in the analysis. The consistency check may help uncover biases and inconsistencies in judgements. Also, the hierarchical way of structuring the problem may help understanding the problem and the value system (Andresen, 2000). The disadvantage with AHP when there are many components is that the number of pairwise comparisons grows with the rate: $n \cdot (n-1)/2$, where n is the number of components. For example, if there are 10 components, there will be 45 pairwise comparisons.

Introduction

AHP is here proposed in the evaluation of decisioning criteria of the early design process with the focus on integration in integrated dynamic models. The method suggested here is the use of AHP in a direct form, minded towards the clarification of design criteria in multidisciplinary environments. Similar to the decision diagrams (Section 4.4.1) the AHP process may be used for clarifying differences between solutions. The AHP has been used to solve complex political crises in the Middle East (Saaty and Zoffer, 2013), therefore one may regard AHP to be sufficient to rate and order conflicting objectives between conflicting parties or objectives in building design.

Table 15 Fundamental Scale of AHP when scaling two components, from (Saaty, 2008)

Intensity of Importance	*Definition*	*Explanation*
1	Equal Importance	Two activities contribute equally to the objective
2	Weak or slight	
3	Moderate importance	Experience and judgment slightly favor one activity over another
4	Moderate plus	
5	Strong importance	Experience and judgment strongly favor one activity over another
6	Strong plus	
7	Very strong or demonstrated importance	An activity is favored very strongly over another; its dominance demonstrated in practice
8	Very,very strong	
9	Extreme importance	The evidence favoring one activity over another is of the highest possible order of affirmation
1.1–1.9	When activities are very close a decimal is added to 1 to show their difference as appropriate	A better alternative way to assigning the small decimals is to compare two close activities with other widely contrasting ones, favoring the larger one a little over the smaller one when using the 1–9 values.
Reciprocals of above	If activity *i* has one of the above nonzero numbers assigned to it when compared with activity *j*, then *j* has the reciprocal value when compared with *i*	A logical assumption
Measurements from ratio scales		When it is desired to use such numbers in physical applications. Alternatively, often one estimates the ratios of such magnitudes by using judgment

Scaling method

The scaling system of AHP works by scaling components in relation to the "intensity of importance", see also Table 15.

A component could be anything of value to the building design, for example High-Performance Building or *beautiful spaces for real people.* The method exposes components in pairs and lets the user evaluate the relative importance of one component to the other. After evaluating all components two and two in the matrix, all the components are automatically ranked after importance. To maintain a reasonable valid scaling of very different components, a subdivision of scale is suggested as the following two-step procedure.

First step towards unified quantification: Scaling the components systematically

Scaling components of multiple quantifiable and unquantifiable factors is divided into separate steps. This is done partially to distribute the process of *scaling* to those in the design team that hold the largest amount of insight within a certain discipline. Another reason is to provide a system that clearly separates which components are regulated – therefore non-negotiable, to those components that are unregulated.

Components subject to scale:

- Decisioning criteria
- Categories (disciplines and areas of focus)
- Type (negotiable legislative factors, non-negotiable legislative factors, optional factors)

Scaling decisioning criteria

Scaling the importance of decisioning criteria is needed to evaluate one solution over another. The base values of decisioning criteria may be quantifiable and subject to conversion but can also consist of quality defined arguments.
An example of two quantifiable decisioning criteria can be *yearly energy building energy consumption below national requirements, kWh/m^2/year* scaled as less important than *overheating hours below national requirement, hours/year.* An example of two quality-defined decisioning criteria can be *physical and mental recreation in outdoor environment* scaled as less important than *peripheral boundaries to provide activities and temporary habitation*.
The scaling of the decisioning criteria is obviously a highly subjective task. This is why the scaling must be done in an environment that provides a maximum amount of transparency and why the scaling must be done by an expert.

Scaling categories – policy-maker's decision

An expert may be defined by the personal/disciplinary background, education or field of interest. What is important is that the expert must be able to define a discrete category in which his or her expertise is defined. Such categories can be defined by various *knowledge fields* or *disciplines* e.g. environmental sustainability, social sustainability or economic sustainability as seen in Figure 63.

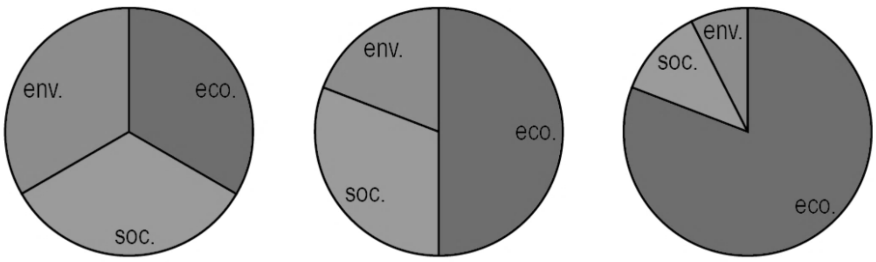


Figure 63 Scaling categories. To the left is a model of equal categorical importance, the middle and the right hand side show other variants of biased categorical importance.

The scaling is first applied to categories and ideally this scaling reflects the client's own wishes. The scaling of categories may also be used as a method of discussing and scaling the focus for the entire project. The method described here will allow the design team to alter the scale of categories during the process of evaluation if the categorical importance shifts during the early design stage. It must, however, be advised to define the scale of categories as early as possible, mainly to avoid interdisciplinary discussions of who, what and which discipline is more important. In a later stage where more than one solution has to be evaluated, all categories and all decisioning variables must be subject to the design team's choice of scaled categories. This is done to maintain a consistent comparison of the different solutions.

Scaling across various types of requirements

The next level of components to be scaled is the importance of criteria within each category. The environmental category may, for example, be composed of energy requirements, requirements relating to daylight and thermal indoor environment. Some of these requirements may be subject to regulations and other non-negotiable parameters (for example maximum cost set by the client). To maximize transparency of the scaling of criteria it is therefore necessary to determine which criteria are defined by the individual member of the design team, and which are defined by external imposed requirements.

Another reason for separating non-negotiable factors[24] from options factors is that it is useful when building solutions need to be compared over different types of legislation. An example is when trying to compare a solution that is defined by the low energy classification 2015 with low energy classification 2020 (Danish Building Regulations, 2013). This was attempted in the project Nordfløjen. Often such comparisons are quite difficult as the benefits and disadvantages of the different solutions are subjects to different laws or different measurement methods (e.g. changing primary factors on electricity use and district heating). Therefore, solutions cannot be compared directly on their energy use or in terms of their ability to provide satisfactory daylight conditions. Nevertheless, comparisons across legislative requirements are something clients ask for. A third reason to separate non-negotiable factors from options factors is to identify solutions that always comply with legislations and imposed criteria. All solutions that do not comply with these can be discarded, or they may be subject to critical discussion on why other criteria are more important. As a consequence, one solution may perform outstandingly in terms of all categories and all optional criteria but need a statutory permit to be built, because the solution violates one criterion. Some clients are willing to account the risk of not fulfilling certain regulations, such as. local politically imposed requirements requiring special permits.

[24] Factors are used instead of criteria, as criteria in themselves are non-negotiable

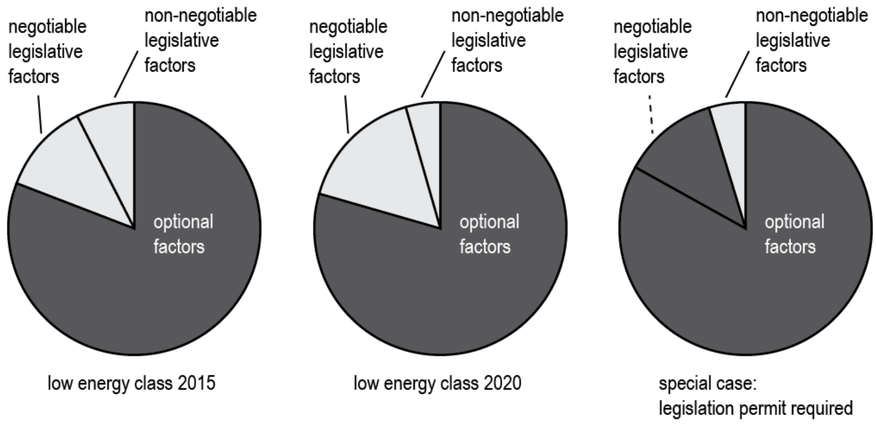


Figure 64 Examples of scaling the negotiable factors, the non-negotiable factors and optional factors.

Negotiable and non-negotiable factors are subject to the same scaling procedure as all of the optional factors (Figure 64). In that sense, fulfilling requirements from imposed criteria is not the same as having a building solution that performs "well". It only suggests that the building fulfills the minimum requirements. To be able to fulfill more than the minimum required by regulations, the design team need to clearly state this as a requirement of their own. This requirement may be more important than non-negotiable factors and should be scaled thereafter. Even regulated requirements may be less important than other regulated requirements when seeking to create e.g. High-Performance Buildings. But they are regulated for a reason. Assumed a regulated requirement defines a minimum WWR in a building, such a requirement can be a good thing for the total energy consumption for daylight etc. But compared to the combined regulated requirements of minimum daylight factor, thermal environment and energy consumption the WWR may seem less important. Legislative factors are for this reason not equally important for the client or the expert scaling and evaluating the requirements. The project Nordic Built among other things showed that the current Danish regulations did not value design choices that lowered the energy use by e.g. monitoring personal energy consumption, neither did the regulations consider any real social benefits. In some cases, the regulations are simply not up for the job to create *beautiful buildings for real people* or as discussed in Section 2, not adequate to create High-Performance Buildings.

Step two: Evaluating the solution(s) - design team

When the criteria-scaling procedure is set, and divided into negotiable, non-negotiable and optional factors, the actual evaluation process of solutions can begin.

This process requires the same amount of expertise as in the first step of scaling the components. The evaluator has to be specific when "grading" each criterion relating to a particular solution. The grading will ideally be backed by an argument, a calculation or a simulation. And in relation to the rest of the thesis, this should be maintained by an integrated dynamic model.

An expert evaluator in building energy would as an example have three criteria in relation to which a solution is evaluated upon:

- Yearly energy consumption LE2020 < 25 kwh/m^2/year
- Access to daylight DF>3%
- Number of overheating hours for T_{in}>26°C

These criteria have been scaled to be equally important in the step before. Two solutions were produced, each criterion is given a point from 1-10 where higher points are better[25]. The two solutions could be evaluated as seen in Table 16.

Table 16 Equally scaled criteria

Criterion	AHP	Solution 1	pt.		Solution 2	pt.
Yearly energy consumption	0.333	22.4	6	>	23.1	4
Access to daylight	0.333	3.1	1	<	4.2	5
Number of overheating hours	0.333	75	7	>	102	4
Better solution (highest weighed sum)		**Solution 1**	**4.66**			**4.33**

If the different criteria were scaled for some reason, for an example *access to daylight* is considered twice as important as any of the two other factors, the chosen solution could be quite different as seen in Table 17. Of course these criteria must be considered along with many other criteria, and therefore which of the two solutions is better, is very much up to the entire team and their choice of scale.

Table 17 Unequally scaled criteria

Criterion	AHP	Solution 1	pt.		Solution 2	pt.
Yearly energy consumption	0.250	22.4	6	>	23.1	4
Access to daylight	0.500	3.1	1	<	4.2	5
Number of overheating hours	0.250	75	7	>	102	4
Better solution (highest weighed sum)			**3.75**		**Solution 2**	**4.5**

Conclusion

Until a way of quantifying any qualitative factor has been found, tested and standardized no definite methodology exists of equating the un-quantifiable with the quantifiable factors. By using the suggested method, different aspects of quantifying premises of social and environmental sustainability become more

[25] This point system could be based on any evaluation method, even AHP itself. Each criterion needs to be normalized for the system to work. Ideally, an integrated dynamic model can handle the translation from numeric values received form BPS tools to normalized points.

transparent. The key issue is being honest in every aspect of the evaluation, from the building programming to the building operation period. Building codes and regulations do not explicitly ensure low energy buildings, better indoor environment, social sustainability etc. Certification systems like BREEAM and DGNB do add to the expectation of a High-Performance Building but they do not cover all of the client's intensions and in some cases they will contradict what the client wants from their building. The suggested method provides a means of evaluating any given idea and transferring that into a concept of quantification based on the design team's own definition. The methodology helps the client and the design team to define criteria and the coherence between these criteria. Furthermore, the method helps the client and the design team to navigate between different proposals and solutions, and in the end it helps to clarify if the client has received what the client asked for. With respect to social and environmental sustainability the method should be able to clarify the decision process better than the qualitative method used in e.g. Nordic Built. Also, the AHP method may help the design team to choose which objectives are more important. This can be used in the pre-modeling step in combination with decision diagrams (see 4.4.1), or to simplify the process of putting weights on multiple objectives when combined with integrated dynamic models with optimization.

4.4.3 Development: Termite

Termite is developed by the author and is a free tool for all non-commercial purposes. It has been used in different ways since it was first developed. It is planned to keep on developing the tool and keep introducing the users to more features. This section demonstrates some thoughts behind Termite and how it works.

Introduction

Termite is a parametric tool using the Danish building performance simulation engine Be10 (SBI, 2013) written for the Rhino-Grasshopper environment. The tool Be10 is originally intended for building energy frame calculations and is required by Danish law (Danish Building Regulations, 2013) when constructing new buildings. Termite opens up for various types of analyses relating to energy performance. This ranges from component level analyses over abstractions of buildings, to entire building energy evaluations. Additionally, fully parametric district and city-size simulations of yearly building energy consumption were shown possible (Negendahl, 2014a) with the same precisions of energy use, since the tool simulates on each and every building. Figure 68 demonstrates some of the parametric possibilities in using Termite e.g. planning for optimal synergetic envelope requirements, placing solar energy production facilities etc.

Termite is able to simulate the dynamics of building energy consumption over the year, which include thermal transport, natural and mechanical ventilation, cooling and heating systems, heat pumps, solar cells and much more (Figure 65).

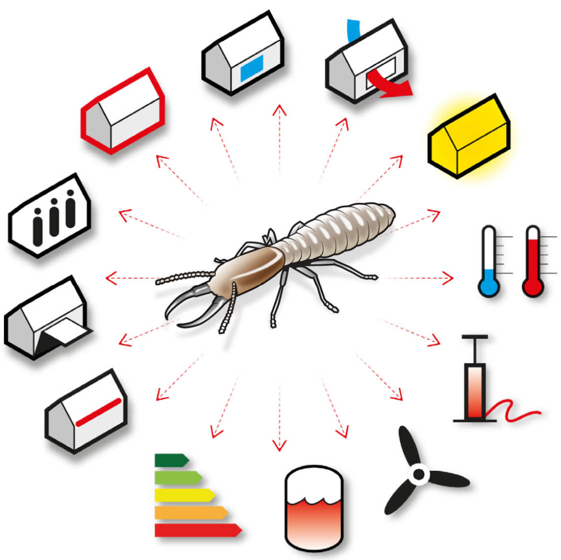


Figure 65 Termite building energy simulation tool

Termite features

- Live energy performance feedback visualized directly in the design tool Rhino
- Can be included in a parametric design process “on architectural terms”
- Is fully parametric and can handle any types of buildings in Denmark
- Uses the simulation engine Be10 directly by run-time couplings thus ensuring a calculating validity that meets the Danish building regulations
-

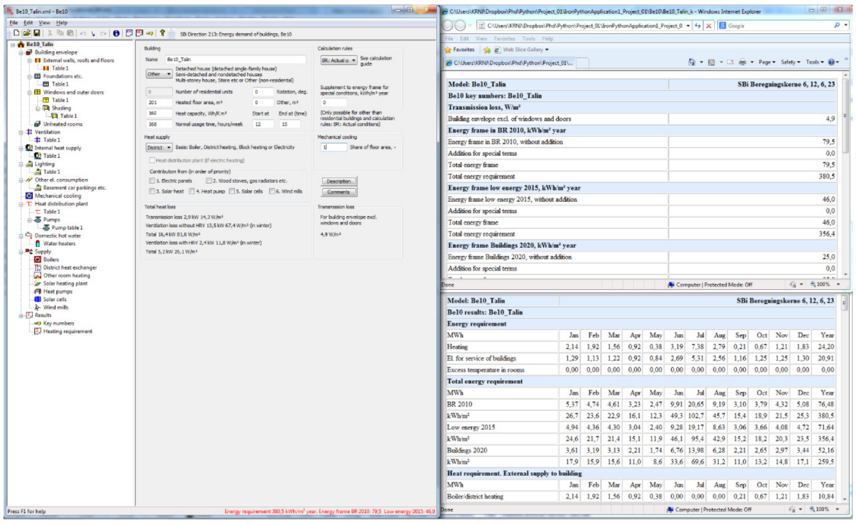


Figure 66 A typical screenshot of Be10 UI. To the left is the spreadsheet layout for input. The results are printed to the right.

Be10 is originally developed to calculate the energy consumption of all new buildings in Denmark. The software is obligatory for governmental approvals for new buildings, and therefore should be used to calculate the energy frame for any project regardless of size. The software is developed by SBI – the Danish Building Research Institute. The downside of the program is primarily its user interface (UI – see Figure 66). The program is difficult to use and not very accessible for non-simulationists. There is no direct correlation between the model in Be10 and a building design, as the entire building is represented in schematic form in Be10. Be10 does not support the import of geometry or other data exchange and cannot be connected to other tools (until now).

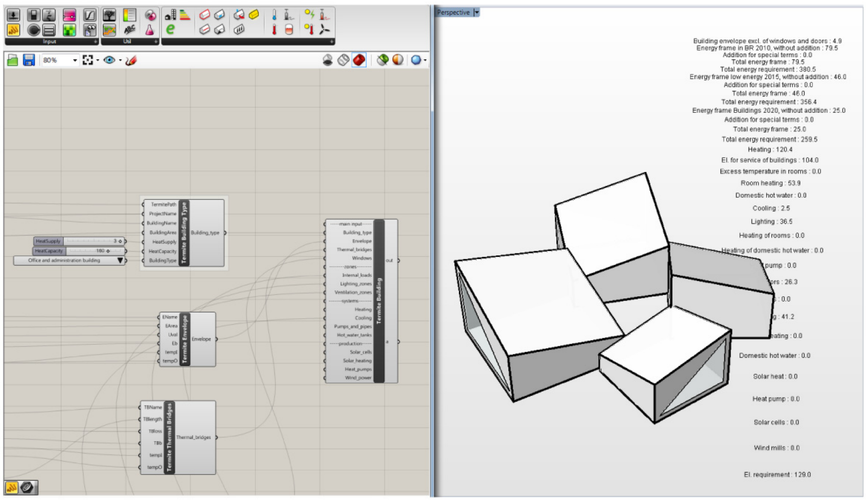


Figure 67 A typical screenshot of Termite UI. To the left is the graph based layout for input. The results are printed to the right.

Termite works simply by using the simulation solver in Be10 in the dynamic sense, so Be10's solver is run-time linked to the VPL and thus the design tool Rhino (Figure 67). All features and any methods that are implemented in Be10 are accessible to the user, not in schematic form but in graph based form. Simple modules written specifically for the early design stage, allows for manual input to be replaced with dynamic links to a geometry model. The results are fed back *live (one simulation takes less than 0.1 seconds)*, and can be illustrated by colors or values directly in a design tool as it suits the designer best.

Figure 68 illustrate the self-shading mechanisms in a fairly condensed city scape. The shading is calculated with an isovist-method based on C. Reinhart insulation distribution used in the BPS tool Radiance (Reinhart and Andersen, 2006). The shadow effects are then used by Termite in calculating the monthly heat gain through window openings.

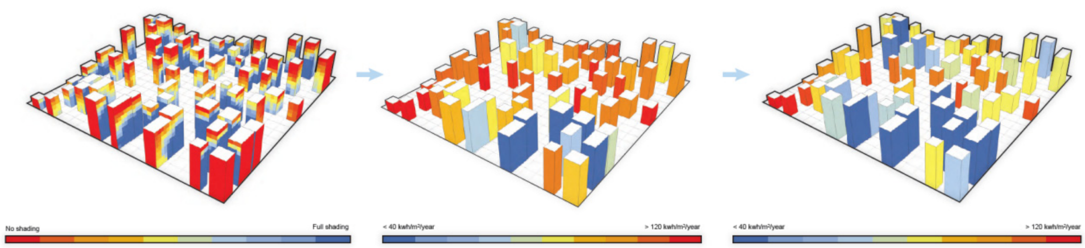


Figure 68 With Termite it is possible to make entire district size evaluations extremely fast. The results may be visualized on the buildings to create a whole new analytical dimension. From Negendahl (Negendahl, 2014a)

Termite can be used to calculate the monthly and yearly energy consumption in kWh/m^2, here displayed as a color of building energy consumption. The entire city site is simulated within 5 seconds on an ordinary desktop machine, thus making the tool ideal for parametric design exploration purposes. Fundamental changes in heating strategies and ventilation requirements can lead to very different energy consumption. Here the district heating exchanger efficiency is improved by 10% and criteria on window g-values are changed in all buildings.

Evaluation of Termite

Termite has been used in several projects some of which are demonstrated in here in the case studies: Kirk, Nordfløjen and Good Year. The plugin has been used by several students and has been given to all the architects which Grontmij have been collaborating with during the Ph.D-project. The plugin has been expanded, altered and modified several times, either to fix bugs or to better support the design teams during or in following projects. There has been no attempt from the author to gather feedback or evaluate the plugin, which is needed to validate the usability, flexibility, speed, etc. Nonetheless feedback has been given from both students and external users (mainly architects). These feedbacks has often assisted in finding bugs or been the reason of new feature additions to the plugin. The plugin is still considered under development and is likely to change. Also the coupled BPS tool Be10 (SBI, 2013) is under continuous development and improvement, therefore the usability, flexibility, speed etc. as well as the overall performance of Termite is destined to change when Be10 does.

Conclusion

Termite is a direct answer to the need of design teams to generate fast and reliable entire building energy simulations. Termite is a quasi-steady-state simulation tool using the solver from Be10. Termite is built specifically to ease the modeling process of integrated dynamic models as it works as a plugin in the VPL Grasshopper.

4.4.4 Development: HQSS

In the following section the HQSS tool is explained. The tool was used in the Good Year project discussed in Section 4.1.5 project and published in Energy and Buildings in (Negendahl and Nielsen, 2015).

Introduction

When considering the risk of overheating only few tools presently can evaluate entire buildings fast enough to effectively be used in early stage design processes. Most of these tools rely on dynamic methods (see 2.4.2) which are slow even they are only considering single zone approximations. The clear limitation of HQSS (as for any quasi steady-state method) is the lack of time depended dynamics. This means true utilization of thermal mass, radiant temperatures, venting and cooling strategies and any system or user dependency of schedules is difficult to take into account. The advantage of quasi steady-state methods on the other hand is speed. The tool is now a part of the Termite plugin for Grasshopper.

The purpose of HQSS tool is a simple evaluation of cooling capacity efficiency on an hourly basis simply by determining the accumulated hours where the cooling capacity $Q_{C,cap}$ does not meet the heat loads Q_{load} at each calculation step t:

$$\sum_{t=1}^{nt} \left(Q_{C,cap} \leq \; Q_{load} \right) \tag{1}$$

where $Q_{\;C,cap}$ is the cooling capacity and Q_{load} is the heat loads at any calculation step t, t is defined as one hour in the range of a year of 8760 hours. However, to speed up the calculation process the number of calculation steps, nt is reduced in two ways. A) Only hours, t within the service period (usage profile) of the given zone are considered, in this case as an office open [08-17] every day, all year. B) Only hours, t where direct solar irradiance has an effect on the given zone are considered, see equation (8).

For each building zone for each calculation step the total heat transfer, Q_{ht} is given by (ISO, 2008):

$$Q_{ht} = \; Q_{tr} + Q_{ve} \tag{2}$$

where Q_{tr} is the total heat transfer by transmission and Q_{ve} is the total heat transfer by ventilation.

The total heat gains are expressed as:

$$Q_{gn} = \; Q_{int} + Q_{sol} \tag{3}$$

where Q_{gn} is the total heat gains for each calculation step, Q_{int} is the sum of internal heat gains, and Q_{sol} is the sum of solar heat gains over the given period. The ideal cooling demand at any point in time where the sum of heat gains are larger than the sum of (positive) heat transfers can be expressed as;

$$Q_{C,nd,cont} = \; Q_{gn} - \eta_{C,ls} \cdot \; Q_{ht} \tag{4}$$

where $Q_{C,nd,cont}$ is the needed amount of cooling to maintain set point temperatures and $\eta_{C,ls}$ is a dimensionless utilization factor dependant on time constants and used specifically in seasonal and monthly calculation periods (ISO, 2008). When the maximum cooling capacity, $Q_{C,cap}$ is known, equation (4) can be written as;

$$Q_{C,cap} \geq Q_{int} + Q_{sol} - (Q_{tr} + Q_{ve}) \quad (5)$$

The internal gains, Q_{int} for each zone k in each calculation step t can be extracted as;

$$Q_{int} = \left(\sum_{k=1}^{nk} \left(Q_{equip,k} + Q_{occu,k} + Q_{light,k} \right) \right) t \quad (6)$$

where k is the zone and nk is the number of zones in the building, $Q_{equip,k} = 6 \text{ W/m}^2$ and $Q_{occu,k} = 4 \text{ W/m}^2$ is assumed constant in every calculation step t (since only the service period is considered). $Q_{light,k}$ is calculated as the interpolated value based on a daylight factor, DF from radiance (see equation 13.) The daylight factor is reduced to; *if* $DF > 3\% = 3\%$ and the effect $Q_{light,k}$ is normalized to fit the range $[0..3]\%$ with the expression:

$$Q_{light,k} = \left(\frac{(DF - 0\%) \cdot \left(Q_{light,max,k} - Q_{light,min,k} \right)}{3\% - 0\%} \right) + Q_{light,min,k} \quad (7)$$

The solar gains, Q_{sol} are assumed to be composed of a direct beam component depended on solar position v, and a constant diffuse component dependant on the sun position in the calculation step t;

$$Q_{sol} = \left(\sum_{w=1}^{nw} \left(\sum_{v=1}^{nv} (\cos(\varphi_v) \cdot I_v + Q_{dif}) \cdot g \cdot b \cdot A_w \cdot FR \right) \right) t \quad (8)$$

where w is the window in a façade and nw is the number of windows in the zone, v is the unique sun vector visible from the window and nv is the total amount of vectors. φ is the incidence angle to the sun vector, and I_v is the correspondent (beam component) effect from the sun. g is the g-value of window pane, b is an adjustment factor, which is further described in the discussion, A_w is the window area and FR is the frame ratio, Q_{dif} is the diffuse contribution calculated to: 70W for the particular site. Q_{dif} is estimated as an average fraction of horizontal diffuse radiation, Dh with the function;

$$Q_{dif} = \overline{Dh} \cdot (180 - \beta/180) \quad (9)$$

where β is the inclination angle of 90°.

The solar gains evaluation is defined as an annual simplified solar beam component simulation. To speed up the calculation process the annual hourly sun vectors are reduced from 8760 to 103 vectors, while the irradiance effect, I_v

per unique sun vector, v is maintained in every vector group. As a consequence, each originally placed vector is repositioned slightly on the hemisphere (see Figure 69). While this will affect the angle of incidence, φ_v, the precision of the calculations are only slightly biased in the process, more on this subject is found in the discussion. If unobstructed, each irradiance factor with the new angle of incidence for each window is calculated. However, most sun vectors are obstructed by the building geometry so most sun vectors are omitted from the calculation; this again makes calculations run significantly faster. The obstruction calculation is processed by an isovist[26] (Benedikt, 1979) function.

The transmission losses/gains, Q_{tr} for each zone k in the each calculation step t are extracted as;

$$Q_{tr} = \left(\sum_{k=1}^{nk} A_{win,k} \cdot U_{win,k} + A_{wall,k} \cdot U_{wall,k} + l_{\frac{win}{wall},k} \cdot \psi_{\frac{win}{wall},k} \right) \cdot \left(\theta_{set,i} - \theta_e\right) t \tag{10}$$

where $A_{win,k}$ is the area and $U_{win,k}$ is the U-value of the window (inclusive frame), $A_{wall,k}$ and $U_{wall,k}$ are the area and U-value of the opaque part of the façade. $l_{win/wall,k}$ and $\psi_{win/wall,k}$ is the length and transmission factor of the connection between wall and window. The cooling set point temperature $\theta_{set,i}$ is assumed $26°\mathrm{C}$, infiltration is ignored and θ_e is the external temperature at calculation step t.

The ventilation loss/gains, Q_{ve} for each zone k in the each calculation step t are extracted as;

$$Q_{ve} = \rho_{air} \cdot c_{air} \cdot \left(\sum_{k=1}^{nk} b_{ve,k} \cdot q_{ve,k,max} \right) \cdot \left(\theta_{ve,set,i} - \theta_e\right) t \tag{11}$$

where $\rho_{air} \cdot c_{air}$ is the heat capacity of air volume set to $1200\ \mathrm{J/(m^3K)}$, $b_{ve,k}$ is the dimensionless temperature adjustment factor representing the heat recovery rate. $q_{ve,k,max}$ is the maximum airflow expressed in m^3/s. The air supply temperature $\theta_{ve,set,i}$ is assumed to be $18°\mathrm{C}$ and θ_e is the external temperature at calculation step.

[26] *Isovist is defined as an object that can be seen from a given point in space*

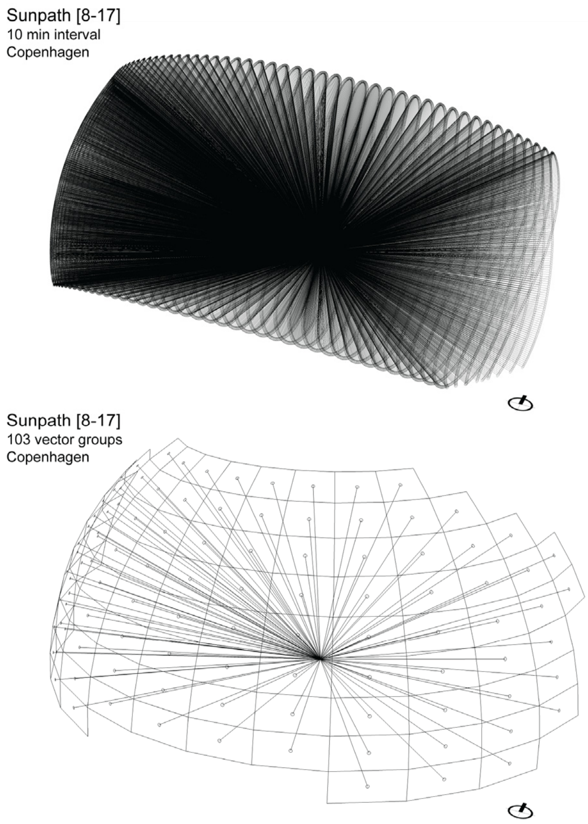


Figure 69 Annual solar sky component generated from the usage profile of a typical office [8-17]. The reduced vector field can be seen in the bottom picture.

Evaluation of HQSS

HQSS has only been used once (Good Year) and was developed for this particular project. The plugin has been given to students who have been using it for further improvement and hopefully a general use in other projects. There has been no attempt from the author to collect feedback on the particular plugin. Therefore its usability, flexibility, speed, etc. need to be verified by further studies. HQSS has been through peer review in a renowned journal (Energy and Buildings) which makes it a promising tool for further investigation.

Conclusion

The HQSS tool is meant to provide very early design stage thermal indoor environment evaluations. A thorough test of HQSS has been documented in (Negendahl and Nielsen, 2015), in which it is compared to Energy+ (U.S. Department of Energy, 2013). Based on these results HQSS is considered precise enough for early design stage thermal investigations and may be used for optimization purposes. However, as the authors state, HQSS should be used with care and only to determine the direction of design, not the final design. HQSS is built to ease the modeling of integrated dynamic models and to provide fast and reliable estimations of thermal indoor environment.

4.4.5 Development: Moth

The following algorithm Moth is developed by the author and part of a student project by Perkov, T. and is described in the paper "Agent-based decision control - how to appreciate multivariate optimization in architecture" (Negendahl et al., 2015). Parts of the paper are used to explain the concept of the optimization algorithm. The reason to include this development in the thesis is to explain some of the efforts in trying to improve the conditions of the inclusion of optimization in early design stages. As explained in in Chapter 3, existing optimization algorithms are difficult to integrate in the design process. Moth is an attempt to address this problem. For more details in the performance of the algorithm please see the paper attached to this thesis.

Introduction

Building performance optimizations during early stages of the design process are not only related to risks of high uncertainty (Lin and Gerber, 2014), but also require an excessive amount of calculations that are very time consuming (Salminen et al., 2012). The early design stage can be characterized by a limited amount of information about the building's architecture and at the same time a high frequency of design changes. In contrast to that, most optimization methods rely on clear objectives and well defined boundaries. Such requirements are rarely associated with the early design stage. The process of designing can be far better described as an exploration of boundaries and objectives rather than finding the one solution inside a fixed number of boundaries and objectives. For optimization to be truly appreciated in building design both qualitative and quantitative objectives as well as "real" human control of these need to be part of the exploration process during the building optimization.

The agent based modeling approach

Bilboria (Biloria, 2011) showed ways to integrate multi agent models combing environmental data and emergent architecture. While the logics behind the various agent behaviors were based on metrological data, expert and engineering consultations, the agents were not directly coupled to simulation environments. What distinguishes ABMs from the more classical stochastic approaches is the ability of ABMs to decompose a global problem into a number of smaller ''local'' problems that may be solved individually and simultaneously (Davidsson et al., 2007). In this way, reducing the size of the search space into single sub-problems could help in the achievement of finding better solutions faster. Agent based modelling often uses the term self-organization. This can be understood as a process where a structure or pattern appears in a system without a central authority (Biloria, 2011). ABMs are also Characterized by being able to adapt quickly to changing environments (often mentioned as high dynamicity / time scale reactiveness/ changeability). This is primarily due to the way agents work in distributed networks in parallel and continuously strive for a state of individual stability. Compared to the classic optimization approaches, ABMs can run continuously with radically changing inputs, where most stochastic methods often need a complete restart.

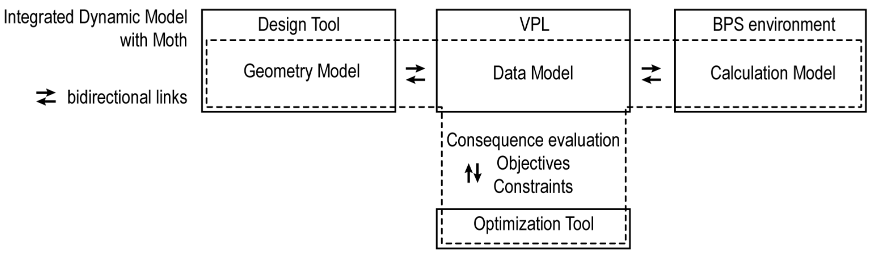


Figure 70 Integrated dynamic model with Moth. Optimization is in-state, and does not require a catalogue of candidate models to predict optimal performance.

How Moth works

The presented method makes use of a new adaptive, open agent-based optimization algorithm named Moth; it has been developed by the authors. The algorithm allows parametric geometry and any other parametric variables to be controlled by individual agents.

Moth stands for Multivariate Optimization wiTh Heterogeneous agents. It is a fully scalable algorithm, thus capable of taking any number of quantifiable objectives with any number of boundaries. Moth is a heterogeneous system; this means that every agent actually comes in opposite pair; a minus- and a plus-agent. Moth is developed for the Rhino-Grasshopper environment (Robert McNeel & Associates, 2013a), and in this way it is integrated in the fast growing parametric universe supported by many enthusiasts and designers using the environment. Moth is open source and built in the IronPython programming language (Viehland et al., 2015).

Each Moth-agent has as a goal to improve its own dynamic objective. The Moth-agents can be manipulated by the model operator (designer) during the optimization process as seen in Figure 72. This gives the designer an option to focus on selected areas of a building (in the Euclidean space). The idea is to give the designer live decision control over a complex system that continuously seeks to find balance in the multivariate decision space.

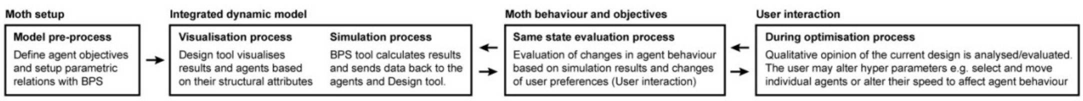


Figure 71 Moth - The agent-based optimization algorithm allows user feedback and feedback from one or more coupled BPS tools facilitated through an integrated dynamic model. No post process is necessary.

The paired agents (-agent and a +agent) seek to reach the same objective by inversely changing design variables. These changes are based on the evaluation feedback from a coupled BPS tool (see Figure 70 and 71). The evaluation of feedbacks is tied to the agent behavior through a *promotion* system. The strongest agent in the pair gets promoted by a positive feedback, thus giving the promoted agent more authority to change its design variables in its own preferred direction (either positive or negative). In this way every agent-pair does not need to have any pre-defined preference or knowledge of what it is supposed to do other than what design variables it is allowed to manipulate. These design variables connected to the agent-pairs are simply native Grasshopper *sliders,* which means that any building designer that is capable of using sliders to control parametric geometry in Grasshopper can with little effort use the Moth-agents as assisting decision support the early design stages.

Moth-agents will be attracted or flock around geometrical features in the model representing the design variables. In the case in the presented paper (Negendahl et al., 2015) windows and a proxy point placed in the center of the building representing the window type are deemed design variables. The number of active agents assigned to a particular design variable will define the focus of the search.

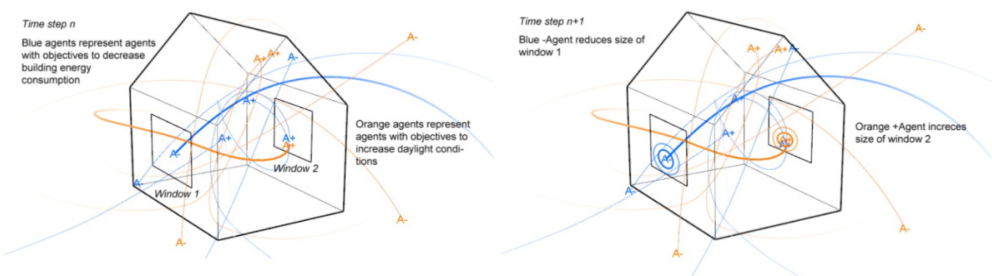


Figure 72 Example of Moth-agents in action.

Figure 72 illustrate how Moth works during optimization. Here, to make the process simpler, two types of the agent pairs and two variations of the building geometry are illustrated. The two types of agent pairs seek to modify the design variables to reach their individual objectives; daylight and energy consumption. The orange Moth-agent objective is to increase daylight by affecting window openings in the building. The Moth-pairs will be attracted towards windows and each +agent will try to increase the window size, while each -agent will do the opposite. Based on the feedback, the +agent will over time get promoted (as larger windows will let more daylight inside the building), and the +agent it will try to further increase the windows sizes. The -agent will on the other hand have decreasing success in affecting geometry, as it will be gradually demoted during the process. The promotion can be explained as the Moth-agents' 'strength of focus' on the attracting geometry - here windows. The +agent will have more success in circling and thus hitting and affecting the windows that provide most daylight by changing (increasing) the size, whereas the -agent will seem to seek other areas to improve its own goals, that is reducing window sizes, with little success. Over time the +agent will increase windows that make greatest impact on the daylight evaluation, this process is considered a *single objective optimization*. When more agents are added to the system, and these agents are coupled to other objectives, such as energy consumption and capital cost functions like the one showed in (Negendahl et al., 2015), the agent-based optimization is scaled with the number of objectives.

Results

A lengthy qualitative test on Moth's performance and a comparison to two evolutionary algorithms (Galapagos (Rutten, 2010) and Goat (Simon Flöry et al., 2015)) have been performed. One of the findings was that the evolutionary algorithms were better in spreading their search in the search spaces, but (in the particular tests) Moth was able to find minima like the other algorithms. The main difference between Moth and any other currently known implementations of optimization algorithms available in the Grasshopper- or Dynamo-environments, is that Moth can be controlled *in-state* during optimization. More on this subject and further discussions on optimization algorithms are found in the papers (Negendahl and Nielsen, 2015) and (Negendahl et al., 2015).

Conclusion

It is concluded that agents may provide valuable design feedback on building performance (e.g. energy, capital cost and indoor environment). Agents may help optimize open ended multivariate design problems. However, the presented system is not particularly efficient in doing so. Given the many fixed and dynamic constraints as well as discontinuities in design variables the algorithm has difficulties in finding true global multivariate optima. However, the presented agent-based system is fully open and adaptable and will allow a high degree of operator intervention during optimization. For this reason these agent-based optimization algorithms such as Moth are found better suited in the decision support of early design stages as in contrast to other algorithms, which do not support operator intervention during optimization. The concept behind Moth may help optimization with integrated dynamic models to better include qualitative analyses. However, as Moth remains untested in real projects, the optimization approach may still need further development.

4.4.6 Development: Sentient integrated dynamic models

The idea of sentient models originated from the project Nordfløjen. Later the idea was made into Thomas Perkov's masters project under the supervision of the author. Finally, some of the concepts were changed into its current form as seen in the paper *Approaching Sentient Building Performance Simulation Systems* (Negendahl et al., 2014). The main idea began as a simple way to couple an integrated dynamic model to a database, and then start brute force simulations of all the permutations of open design variables. In projects like Nordfløjen where hundreds of individual rooms may need to be assessed separately (i.e.. daylight and thermal conditions) the model would over time generate all variations of zones in the building and store the results in the database. From the user perspective, the user would not care if the result came directly from the BPS tool or from an intermediate database. However, pulling results from the database was instantaneous regardless of the origin of BPSs. Therefore, in principle it was possible to pre calculate any option in the solution space with any type of BPS tool and deliver the results instantaneously to the design team. This, however, generated a whole new set of challenges.

Introduction

Two different approaches of linking BPS with design tools are dominating. The first approach is coupling highly detailed and complicated BPS environments to the design tools. These systems may be able to calculate the performance to a very precise degree, well beyond the information level of a building design in its conceptual stages. In general, these BPS tools need large computing capacity and will need long time to simulate. The first approach for this reason is often much slower than the second approach and in some instances such a system will block the dynamics of the design process. The second most dominating approach seeks to maximize the responsiveness by either linking simplified BPS or implementing user defined scripts acting as BPS (Klitgaard et al., 2006). Ideally, the right implementation and powerful computing power will allow super-responsive live performance feedback from the BPS. This approach lacks precision and may in the worst case make performance evaluations on incorrect assumptions that again can lead to the very opposite of an improved building performance.

Souza (2012) argues that the validity of modeling and calculation assumptions depends not only on the level of competency but also on the purpose of modeling. In this sense, a good model depends enormously on the experience of the modeler, which comes from practical knowledge and contextual understanding of the subject in order to solve similar problems. Valid operation and BPS tool input requires competent simulation experts or "simulationists" as Souza calls them.

A sentient integrated dynamic model

Sentient, also meaning "*conscious"* and "*responsive"* is a term used for enhanced integrated dynamic models that is able to observe and react "consciously" on user requests. The sentient integrated dynamic model was at eCAADe presented as a “Sentient BPS system” as the authors wanted to state that VPLs are not necessarily *needed* to create user-conscious feedback processes. The proposed concept was nonetheless a special case of an integrated dynamic model. It was found a responsive alternative to simple (*designer friendly,* see Section 2.4.1) BPS tools or which could match the more complicated but slow BPS tools (e.g. computational intensive tools using ray tracing methods or CFD methods). Specifically, this concept was targeted at design team in the early design exploration. The model included parametric modeling procedures, which decreased the decision space into a finite size, see Figure 73. By utilizing a database structure combined with a multivariate interpolation algorithm (through MATLAB) it was feasible to simulate fewer solutions and still provide the building designer with fast and precise results (from one or more building BPS tools).

Basically, the integrated dynamic model had a result database containing building performance feedback data needed to accompany the designer's own solutions. The model was able to reduce the number of solutions needed to be simulated, as *it observes user activity and adjusts the BPS tool to simulate and improve interpolation precision* (Negendahl et al., 2015). To effectively do this, the system attempts to predict the space of interest of the building designer while utilizing multivariate interpolation capabilities of the model. Essentially, the model presents building performance feedback of solutions that is of interest to the designer for decision making in the early design stages, in a very efficient way.

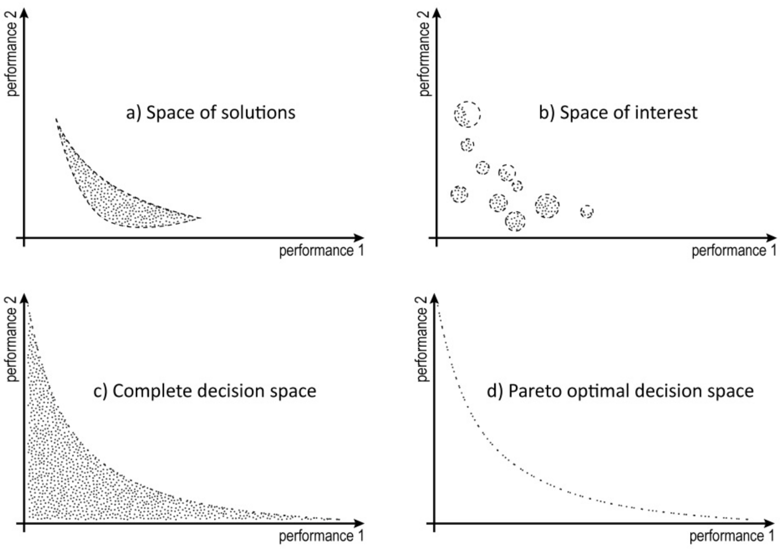


Figure 73 Decision space: Here illustrated in relation to two performance metrics, each dot represents a specific solution. Space of solutions a) is defined by all solutions that conform to the requirement of a certain performance criterion, here performance 1 and 2. Space of interest b) is defined by the building designers’ interest in certain solutions related or unrelated to the performance criteria 1 and 2. The complete and pareto optimal decision spaces c), d) are shown for comparative purposes.

Result database

It was first suggested by Sullivan et al. (1988) that a large number of building energy simulations saved systematically in a database could provide fast feedback on energy performance. Such a database is capable of giving responsive answers to multiple criteria but required either very large databases or very simple buildings to get meaningful answers. Caldas (2001) notes that these kinds of approaches generate data that do "only apply to solutions that are close to those simulated, what makes them of limited use in an architectural design domain". Nonetheless, since the time when Sullivan's and Caldas' considerations were written, much development has been done in the field of databases and computing in general. It may still not be feasible to construct universal databases, comprehending every thinkable combination of variables. But it can be feasible to make a finite subset of solutions as a database lookup that takes a very specific design concept into consideration.

Predictions in the space of interest

Predictions of building designer interest are a rather unexplored subject while predictions of the (space of) solutions have been thoroughly investigated e.g. by (Pedersen, 2006; Shi and Yang, 2013). Framing the space of solutions is defined by very accurately defined objectives, and in terms of building performance, the objectives have to be defined in a way that BPS tools can understand. Predicting the user interest is very different, simply because the user often does not know what he or she is interested in to begin with. The objective is an exploration in itself and consequently objectives are likely to be unclear and fuzzy. The concept is to utilize embedded information of the parametric variables present in the integrated dynamic model. The amount of variables and their *resolutions* will define the amount of unique combinations in the model as it was mentioned above.

A variable resolution is the amount of unique states that a given (parametric) variable has. The variable is an enumeration of numbers, which does not need to be sequential or based on integers. To better control the design variables *variable resolution levels* (Negendahl et al., 2014) were introduced. This helped to further reduce the amount of solutions needed to be simulated. The idea is to make precise performance simulations on strategically selected solutions within the space of interest, then estimate the rest of the space of interest with a minimum amount of errors. The *variable resolution level* of any given sentient model is basically all the unique combinations of every variable state divided by the number of finished simulations (per BPS tool), defined as follows:

Let the resolution r > 0 and the amount of variables v > 1
For every resolution r in the range of variables v_i

$$\text{Variable resolution level} = \frac{r_{v1} \cdot r_{v2} \cdot \ldots \cdot r_{vi}}{\text{finished simulations}} \qquad (12)$$

Essentially, the variable resolution level indicates how much of the space of interest has been covered by simulated results. A high variable resolution level means that few simulations are completed by the coupled BPS tool (in relation to the total number of potential solutions), while a variable resolution level = 1 means that every possible variable combination has been simulated. The number of variables and their resolution will affect the variable resolution level quite substantially. An ideal model will have a minimum required number of variables, each with the lowest possible variable resolutions to cover the space of interest quickly in the design process. Minimizing variables and resolutions, however, can be rather difficult when the building designer has not yet decided all the design objectives. For this reason an *interest prediction algorithm* has been implemented, hence to further reduce the needed simulations, to cover the actual interest space within the boundaries of the defined variables and their resolutions. An interest prediction algorithm is implemented on the basis of a continuous weight factorization of the yet-to-be-simulated unique data combinations.

There are basically three weight functions in the prediction algorithm; *s*, *t*, *w.* They are discussed in detail in (Negendahl et al., 2014), however one of the functions is mentioned here, as it is central for the sentiency of the model. Weight-function, *t* was implemented as a *variable listener function*, which essentially is a timer function that reads the particular variable state of Grasshopper sliders. Basically, the listener function identifies the state of every variable and how long time it remains in that state. The function assigns weights to the design variable with the fewest alterations, which the authors argued "*must be the preferred state of interest of that particular variable"*. The reasoning is that changing design variables are "unwanted" and unchanged design variables are preferred. This is a sound, but flawed argument. First, it is assumed that the designer has defined the parametric model to be as clear and flexible as theoretically possible (see Section 2.1.4), and it is assumed that the designer has created more than one design variable which is controlled by a "slider". These design variables are very likely to be dependent on one another. Hence, a change of one design variable may affect the other variables. Therefore, it is difficult to assume that "preferred variables" are untouched variables.

Conclusion

The whole idea to bring sentiency into the model was shown possible in one particular way: it is fairly easy to develop advanced AI-inspired concepts of this kind with integrated dynamic models. The platform that holds a VPL is very strong in supporting software experiments. Even small ideas can generate great challenges, which many students (and researchers) find impossible to handle in typical BPS tools. The integrated dynamic models was found to be of great value for the development of new concepts such as this particular idea of sentient models. Whether the sentient integrated dynamic models are applicable in reality is difficult to say. First the design team that operates the model has to sufficiently adept in VPLs and automation. Second, the building designers need to be willing to dedicate their time to explore the design choices within the model. And third, these kinds of models are likely to fail if the design changes beyond what the model is capable to within the boundaries of design variables.

4.4.7 Development: Surrogate models

Introduction

In recent years, there has been advancements in the field of Artificial Intelligence (AI) in almost any branch of research. It has spread beyond the academic world with major players like Google, Netflix and Facebook creating their own research teams. This has not yet gained the same momentum in the building industry. Not surprisingly there have been several speculations in how AI can help building design in the industry in practice to improve buildings in general e.g. (Bento and Feijó, 1997; Ehrich and Haymaker, 2011; Parmee and Bonham, 2000; Sanyal et al., 2013). Some of this research can be accredited to researchers who seek to improve building performance, for example (Dong et al., 2005; Eisenhower et al., 2012; Georgescu et al., 2010; Kalogirou, 2000; Pauwels et al., 2011; Qian et al., 2006; Yezioro et al., 2008). Hopfe et al. (2012) used surrogate modeling techniques to approximate the objective functions on energy consumption and over/under-heating hours. The method used Gaussian processes (GPs and sometimes called Kriging), which correlate quite strongly with the introduced noise on the design parameters, to model real-life uncertainties. Other popular approaches such as Linear regression (LR), Support vector machines (SVMs), Tree classifiers and regression (TR) and artificial neural networks (ANNs) can be mentioned.

Almost all BPS tools take too long time to simulate to be effectively used in the design exploration in the early design stages. If consequence feedback is to have an effect on design choices during the early design stages, the feedback has to be reduced to a matter of seconds and not minutes, hours or days. To solve this challenge surrogate modelling techniques to predict "un-simulated solutions" in the vast space of interest are investigated.

Surrogate models in research and practice

The idea to use machine learning with integrated dynamic models is to increase speed and validity. Basically, a machine learning layer is inserted into the model and it acts as a BPS tool, much like the concept of sentient models. The whole point by using machine learning is however not so much to listen to the designer, but to speed up the feedback.

Research within optimization has shown that surrogate models (also called meta-models or emulators) can approximate the original simulation model typically originating from one BPS tool. It mimics the behavior of the simulation model to be able to produce the model responses at reduced computational cost. Surrogate models are often built through an iterative process with a repeated number of evaluations until the desired model accuracy has been achieved. The number of evaluations needed to train the machine learning algorithm is *probably bigger than or the same as the number needed when an optimization algorithm is been coupled to a building simulation engine* (Machairas et al., 2014). Expert knowledge on how artificial intelligence works is essential as well. These are the main reasons why surrogate models are uncommon in building design research *and has a long way to go before it starts being used by practicing professionals* (Machairas et al., 2014). Surrogate models have helped in solving many types building control problems, which benefits from the immediate result feedback. In addition, surrogate models are useful in solving demanding building design problems, with numerous design variables, many local minima and a huge search space (Machairas et al., 2014).

Establishing a surrogate model often goes through three major steps (Nguyen et al., 2014) as follows:

- Sampling input vectors and calculating corresponding model responses, which constitute a database for training a surrogate model
- Constructing the surrogate model based on the database by selecting an appropriate method
- Validating the model before using it as a surrogate of the original model

The second and the third steps may be repeated iteratively until the validation achieves success. Most surrogate models in research used for optimization are set up as seen in Figure 74.

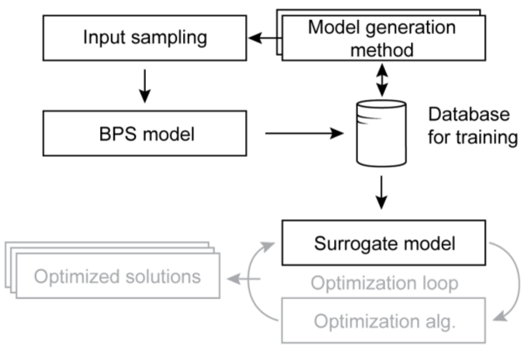


Figure 74 Surrogate model and surrogate model optimization (optional in gray), based on Nguyen et al. (Nguyen et al., 2014)

Testing various surrogate models

The basic idea is to utilize high precision BPS tools but provide near instantaneous performance feedback directly in the design tool. The design tool will request feedback estimations from the surrogate model rather than directly from the BPS tool. To test if integrated dynamic models can substitute the BPS with a surrogate model a simple design case is constructed based on the following assumptions:

- A singular performance feedback is of interest to the designer, for example kWh/m^2/year
- The designer is interested in different types of design variables such as window size, building orientation, ventilation rates, shading geometry, etc.
- The designer does not consider any limitations of the machine learning algorithms being used in the surrogate model
- The designer trusts the performance feedback as if it was made by a BPS tool

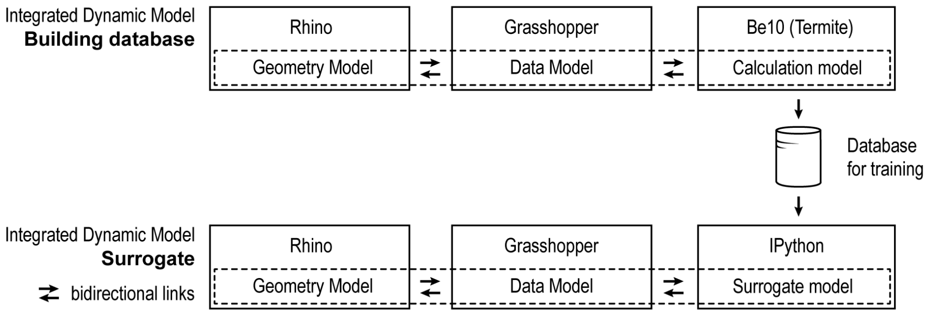


Figure 75 Integrated dynamic model with surrogate models for simulation models

The surrogate model is built in such a way that it is continuously improved by more data from the coupled BPS tool. This is implemented much like the sentient model seen in 0, only this time IPython is used instead of Matlab. Like sentient models the predicted values and the actual values are approaching, $\hat{y} \rightarrow Y$. This idea is represented by using changing sizes of a fixed dataset generated from BPS output from Termite (Be10). Three types of widely used regression methods implemented in Scikit-learn and GPy are tested for the purpose:

· Linear regression (LR), Scikit-learn
· Tree regression (TR) , Scikit-learn
· Gaussian processes (GP), GPy

To get an idea of the performance of the LR, TR and GP, methods have been tested individually on low dimensional datasets. In this section, the methods are only presented for one higher dimensional set, which basically follows the assumptions of a typical early design stage building, as stated above. A 12 dimensional dataset is constructed. There have been no considerations in separating continuous and discontinuous design variables and no considerations in separating high sensitive design variables from low sensitive design variables, as the integrated dynamic model was supposed to reflect a typical early design stage building. To test the methods, the model generation method and Termite have produced a large dataset with around 3000 simulations, all of which have 12 features representing vectors for each design variable.
To begin with, the data has been split in such a way that 15% of the original set is kept hidden from the learning algorithms. This part was used for verification and the rest is used for testing purposes. The remaining 85% is then further sampled gradually. This is done to reflect a growing database generated from a continually refining process from the integrated dynamic model as seen in Figure 75, top.

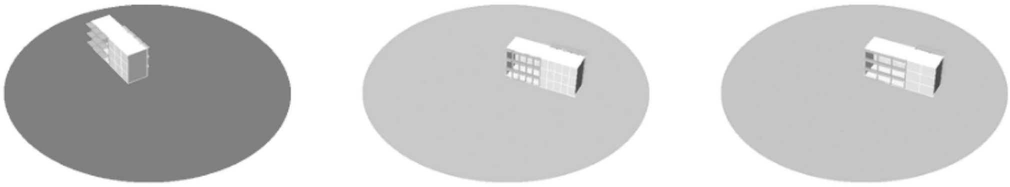


Figure 76 The illustration shows three variations of the building. Model generation is done in an integrated dynamic model with 12 open design variables. The target vector is the simulated energy consumption (kWh/m^2/year).

The simulated energy consumption (kWh/m^2/year) is the *target vector* and 12 different design variables are *the features*. Three randomly chosen combinations of the 12 features are seen in Figure 76.

Three examples of design variables are shown in Figure 77 :

· Rotating a building [0-360] [degrees]
· Window-percentage [0-100] [%] on one side of a building
· Ventilation rate [1-5] [(m^3 /hr)/m^2] of all rooms in the building

It is seen that the effect on building energy consumption is very different. When the design variables are combined with the 9 other design variables (such as U-values of windows and walls, g-values, external shading dimension and room height), the data is impossible to visualize and individually they are of no interest to the building designer.

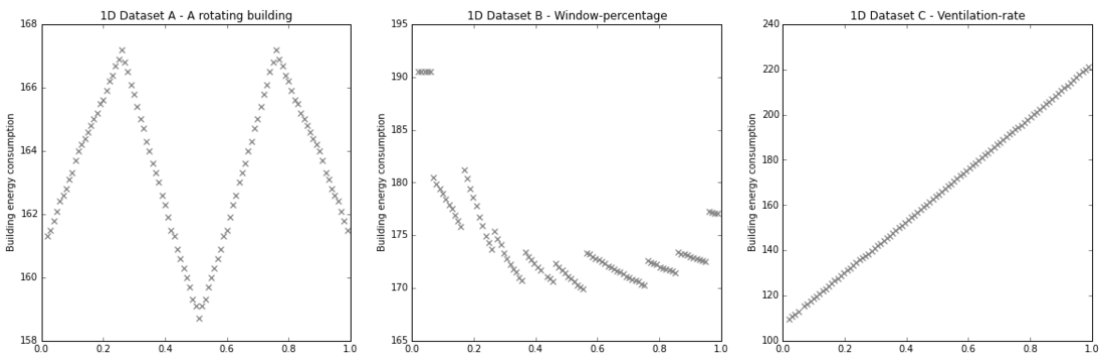


Figure 77 Three examples of the effect of changing design variables on the building energy consumption. Left, rotating building normalized degrees [0-360][°]. Middle Window percentage normalized [0-100][%]. Right ventilation rate normalized [1-5] [(m^3/hr)/m^2]

Linear Regression, LR

Assuming that the regression function $E(X|Y)$ is linear (or at least that the linear model is a reasonable approximation) the following basic form of linear regression model may be used:

$$f(x) = \beta_0 + \sum_{j=1}^{P} X_i \beta_j \tag{13}$$

, where X_j is normalized design variables from the BPS tool

Tree Regression, TR

Different from linear models like LR, logistic regression or SVM, gradient boost trees can model non-linear interactions between the features and the target. TR models are suitable for handling numerical features and categorical features with tens of categories but are less suitable for highly sparse features (such as text data). TRs has the form:

$$f(x) = \sum_{m=1}^{M} c_m I(x \in R_m) \tag{14}$$

, where *M* is the partitions in the regions $R_1, R_2, \ldots R_M$

Gradient boosting methods have been used on the TR, these are also known as Boosting Trees with numerical optimization. Essentially the idea is to utilize a loss function criterion, which is to be minimized. In this test least squares ½$[y_i - f(x_i)]$ and least absolute deviation $|y_i - f(x_i)|$ as loss functions have been used. It should be noted that these functions are solely used to order the information of the input variables. The following loss function $f(x)$ is used to predict y on the training data:

$$L(f) = \sum_{i=1}^{N} L(y_i, f(x_i)) \tag{15}$$

In each stage a TR is fit on the negative gradient of the given loss function. The TR builds an additive model in a forward stage-wise fashion that allows for the optimization of one of the pre-set loss functions.

Gaussian Processes, GP

A Gaussian process (GP) is a distribution over functions. It is fully specified by a mean function and a covariance function: $f \sim GP(m, k)$. The GP provides a prior over an infinite dimensional function. When the covariance matrix is computed using GPy's kern.K(X, X) the covariance matrix between the values of the function corresponds to the input locations in the matrix :

$$f(x) \sim GP(m(x), k(x, x')) \tag{16}$$

This particular GP model is based on a Matern 3/2 covariance function that is built by first defining a covariance function and then combining it with the data to form a GP model. The Matern 3/2 kernel has a distinct exponential part that is beneficial for fitting in minima and maximum:

$$k(x, y) = (1 + |x - y| \times \exp(-|x - y|) \tag{17}$$

Results

LR and TR:
The size of the 12D training set: (449, 12) (449, 1)

GP:
1. The size of the first training set: (29, 12) (29, 1)
2. The size of the second training set: (449, 12) (449, 1)
3. The size of the third training set: (1497, 12) (1497, 1)

The LR, TR and GP models have been exposed to a *gradually growing* test data set. The test set of the size 449 is visualized for LR and TR in Figure 78, top. And the test set sizes 29, 449 and 1497 for GP are visualized in Figure 78, bottom. The vertical axes are the predicted values from the surrogate model and the horizontal axes are the simulated values. The LR is the fastest model to fit (0.02 seconds). The TR fit performed with a grid search with cross validation took 3 min to fit. The GP took 1.7 seconds to fit (to the largest dataset). All models were instantaneous in receiving “look-ups”, that is when giving the model a combination of the 12 design vectors, all models respond with a prediction in less than a millisecond.

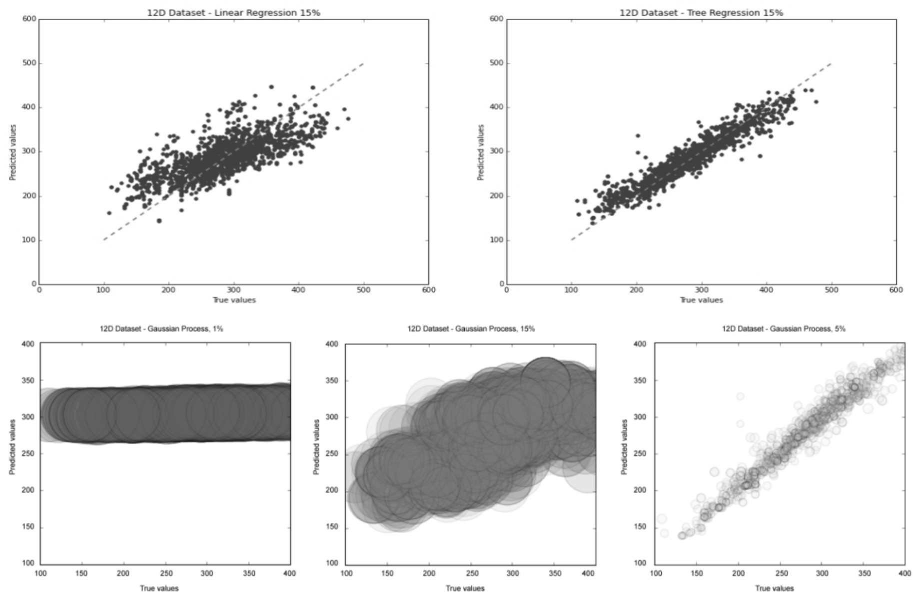


Figure 78 Top left LR trained on 449 data points. Top right TR trained on 449 data points. Bottom, from left GP trained on 29, 449 and 1497 data points, the size of points represents the variance of each prediction.

It can be seen from Figure 78 that a fairly small dataset of 449 data points creates rather different quality fits with different types of machine learning algorithms. The TR is the most reliable of the three methods. Based on this particular dataset the best fit that has been made with any of the models is by using TR with 1100 data points (above this number over-fitting begins to occur); this model is able to predict 60 % within a variance of +/- 10 kWh/m^2/year. This level of precision is far from satisfactory for feedback of energy requirements. In terms of GP it only performed slightly worse than the TR model, but it needed more data points to reach this level of reliability. A unique feature of GPs is that the “inner variance” per data point is known by the model. This depends on the dataset size as seen in the sizes of data points in Figure 78. The danger is to use the variance as a confidence interval to inform the designer *how far off* the prediction is from the suspected values. Over-fitting the model (as seen in the Figure 78, bottom, right) will not only predict wrong results but also wrong a confidence intervals.

Conclusion

Linear regression (LR) and Tree regression (TR) with gradient boosting as well as Gaussian Process (GP) have been tested upon different sizes of datasets. The datasets originate from an integrated dynamic model including the BPS tool Termite/Be10. LR provides very precise prediction accuracy when the data is linear, however most data from BPS tools would be far from linear. The TR method provided a far better accuracy, but the method comes with limitations. TR does not fit well to linear data, they need quite a long time to fit a model and they need rather large datasets for larger dimensions (how much is still an open question). GP provides powerful capabilities in multivariate regression. In simple 1D cases, the precision is very high and model-fit is near instantaneous. The 12D scenarios, which represented a typical building design model in the early design stage, showed some of the weak points of GP models. They are difficult to scale as they need much computing power (grows cubic with the size of data set). The amount of data required by the GP model to fit 12 dimensions are slightly larger than required of the TR model however GP was much faster than the TR. Nevertheless, in terms of the “necessary data size" for these types of surrogate models can be very hard to determine.

As to the question whether surrogate models provide "good enough" accuracy to make meaningful predictions for the early design stage, the answer is no. This is the case at least for the particular test case testing LR, TR and GP, with the condition of a maximum variance of +/- 10 (kWh/m^2/year). Here only around 60% of the predictions were near enough to the true (simulated) values. In real use a building designer might accept a variance of +/- 1.0 (kWh/m^2/year), which is hard to see happening, at least with the methods of LR, TR and GP analysed in this test case.

The above tests have showcased the danger in using auto regression (machine learning) models as surrogate models in the early design stage. When fast feedback from any type of BPS tool is needed, surrogate models might possibly be used. However, to create reliable surrogate models requires careful handling of design variables, careful choice of machine learning methods and careful understanding of the surrogate model output. Therefore, a systematic use surrogate model needs to be defined in a way that ensures that any type of building design variable can be used as a feature and any performance feedback can be used as target vectors.

In terms of feedback speed Linear regression methods, Boosted Tree regression and Gaussian Processes all deliver the same instantaneous results from any type of building performance. Nonetheless, more research is required to determine the better choice of machine learning methods.

4.5 Discussion

In some projects integrated dynamic models were called methods, platforms, systems, tools or toolboxes and they have even been called software programs. The reason for this confusing naming convention is that the integrated dynamic model has a very broad scope in use and application. Integrated dynamic models have been used in several types of projects and performed a multitude of tasks. Not all of these tasks have been considered central for the approach of Consequence based design when first defined back in 2011. Nevertheless, as the project evolved new ways to use the models were developed. The use of more advanced machine learning methods was in the beginning of the project just speculations, but today the author sees the developments of "more intelligent" integrated dynamic models as a natural progression of the presented research. Another interesting aspect is that many of the examples shown in this thesis are direct answers to needs identified in practice. The way the integrated dynamic models are structured around a *design tool* and a *visual programming language*, and the building *performance simulation tool*, have brought more focus to the design process and the process of creation of alternatives than the previous focus on evaluation process of building performance. In the above sentence the words "*design"*, "*visual"*, and "*performance"* are emphasized. These three words in many ways specify the real value of the Consequence based design approach, and wraps the whole point of the approach: to visualize the design and performance. In retrospect the author now sees that the choice of naming the models was based on his background as an engineer: "Dynamic" was about the parametric abilities of the model and the way tools are linked, "integrated" was about the team, the objectives and the cross disciplinary tools and methods. Maybe the integrated dynamic models should have simply been called *visual design- and performance models.*

Nevertheless, the words do still not seem to encapsulate the use of the models. And the author believes that the following years will bring many interesting building projects, which in the early design stage will be formed with integrated dynamic models in some variation of the Consequence based design approach.

4.5.1 Comments on tools

In terms of the design tool and the VPL, the most widely used combination has been the Rhino-Grasshopper environment, however in the past year the competitor to McNeel, Autodesk has put a large effort into the development of the VPL Dynamo for Revit. This author has no preference whatsoever of platform, and will continue to argue that integrated dynamic models can be made with any design tool that has runtime access to a VPL.

In terms of BPS tools used throughout the projects, Be10 has been used extensively, Energy+ and Radiance have in the later projects gained more attention as they were implemented as plugins (Honeybee/Ladybug), and therefore a large acknowledgement goes to the community and developers of these plugins. Daysim and Ecotect have also been of great value to several projects and the same acknowledgements go to the developers of these tools. The interesting part here is that there is no paradigm in the use of open source BPS tools, and integrated dynamic models may couple both open and closed sourced tools. What remains central is that the tools generating most interest in the communities and in the design teams are the tools that are available and open (enough) to be accessed through runtime calls. Therefore for future reference, developers of BPS tools are hereby encouraged to open up their tools for API access or even develop their own plugins for better integration in popular VPLs.

4.5.2 Building performance

The Consequence based design approach has now been used in several projects by Grontmij. As all consulting engineers this company's staff finds satisfaction in providing "optimal" results within their own field of expertise. The engineers in the company have found that integrated dynamic models have given them *more time to analyze* and thus *the ability to provide means of better and faster consulting*. This however is not the same as to have improved the building performance. There is no obvious method to measure if the Consequence based design approach will improve building performance. But many documented cases indicate that it is very likely to do so.
In an internal report the leading simulationist of light and daylight claimed that the approach allowed her to model *much more of the building far more detailed than before*. And the ability to implement fast parametric variations (of rooms) made her *up front with the architect's questions*, even in very complicated geometrical buildings. Whether or not this applies to all "simulationist disciplines" (structural, fire, etc.) has not been investigated, however in terms of energy simulations and thermal indoor environment, there are many indications that the building performance has been improved with the use of integrated dynamic models.

4.5.3 Embedded knowledge

"It is worth reflecting that if powerful, general design algorithms existed, there would be little need for architects or engineers." (Mitchell, 1977)

With the many types of integrated dynamic models, and the many types of tools that are coupled into the models, one might think that these models are very different. However this is not the case. The models are very similar in terms of structure, and many of the rule based methods, such as measuring the WWR and sorting various components are almost exactly the same throughout all the case studies. The reason is found in the "embedded knowledge" that these models collect over time. Methods, rules and even system input such as "average hot water consumption" or "average people load" are almost identical from project to project. It may seem like a poorly designed model, if such variables have not been optimized or altered at least. But the truth is that these repeats have been considered changed, but are almost newer changed. The embedded knowledge may be used in a much more direct way. It can be used as base inputs for unknown factors in the earliest design stage, or this embedded knowledge may be used by disciplines that do not entirely suffice in terms of the quality assurance within the embedded knowledge.
Building designers may in time be able to use integrated dynamic models completely independently of simulationists. The models may have sufficient

knowledge embedded to inform the building designer of what type of analysis is needed, and it may even automate this process. Also simulationists may in time be able to detach the building designer from the entire design process, as the whole framework of building design is gradually embedded in the models.
Exactly 40 years ago Mitchell (1975) performed a survey with the practice of architecture on the topics of automated design. He concluded:

"It does not appear likely that we will discover some basic, underlying "secret" of effective automated architectural design. Rather, the power of effective systems will reside in their capacity to access exceedingly numerous procedures, each of which efficiently performs some relatively small, well understood, and well-bounded task." (Mitchell, 1975)

Still today this seems to be true, and therefore the prospect of architecture as a discipline is far away from being replaced by automations and intelligent models. Whether or not the opposite can happen and embedded knowledge in models and tools will replace simulationists in the near future, is maybe a more likely scenario.

4.5.4 Comments on the commercial aspects of integrated dynamic models

As this project has been developed with aid from the consulting engineers Grontmij the commercial interests in Consequence based design and the use of integrated dynamic models have mainly been driven by this company. The company largely sees Consequence based design as a methodological approach to improve the rational foundation of collaboration between the architect and engineer through the extensive use of integrated dynamic models. As Grontmij's main client[27] is the architect, the buyer-seller relationship is strong between the consulting architects and the consulting engineers. Therefore it is important for consulting engineers to consult the client in the client's interests. The challenge to remain focused on the objectives of High-Performance Buildings is that the end client, the building user, is rarely the architect. Many considerations regarding each individual interest of "clients" on the path to the building user have to be made (see
Figure 79).

[27] Consulting engineers have many other clients, but in the Danish building design industry, the architect is usually the main client who refers to the building owner. In contrast to consulting engineers architects also have stronger relationships to building users. Consulting engineers usually also have a steady income flow from building maintenance and renovation projects, and are therefore compelled towards reaching the clients of building maintenance. These clients in many cases refer back to architects and contractors.

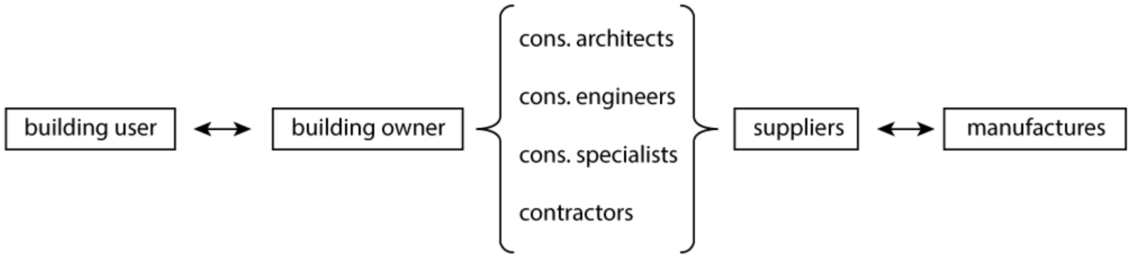


Figure 79 The typical paths through technological innovation system (Bergek et al., 2008)

Consequence based design seeks to include these considerations by striving for High-Performance Buildings in a way that the building designer (the main client) can support and take advantage of performance feedbacks as discussed in Chapter 2. The economic incentive for the consulting engineers is quite clear: if the main client (the architect) sees the benefit in using integrated dynamic models, he might choose Grontmij over the competition. However, to make architects see the benefit in integrated dynamic models is not always as easy as it may seem. The presented five case studies showed many versions of integrated dynamic models, and in all projects the clients saw a clear benefit in the approach. The main challenge is not to convince architects to strive for High-Performance Buildings, but to dedicate the design process towards an evaluation heavy approach. The Consequence based design approach will not work without continuous performance evaluations and the changes to the building design based on consequence feedbacks from the integrated dynamic models. If the building designers are not interested in computational modeling, the approach is very difficult to apply in practice, but if they are uninterested in consequence feedback in early design stage, the approach is impossible to apply in practice. Integrated dynamic models have for these reasons been used in other contexts within Grontmij. The company has used the models as any other BPS tool. This includes; manual modeling, singular criteria definition and traditional reporting of results to the clients.

This process is a step backwards in the perspective of improving the architect-engineer relationship and it is not an efficient approach to create High-Performance Buildings. However, what remains interesting is that integrated dynamic models are used even though the client is not involved, and there is only one reason for this: it pays off. The case study Kirk showed opportunity in improved precision, improved speed and improved means of communicating the results internally. An internal report suggests on average 25%[28] of the resources put into early design stage consulting can be saved if integrated dynamic models are used over traditional BPS tool-approaches. Grontmij argues that it is not uncommon to make changes to a simulation model 8-10 times in the early design

[28] 25% saved resources is based on a comparison of various types of other BPS tools Grontmij uses in the early design stages. The benefit of integrated dynamic models is their parametric capabilities which save the simulationists much time for remodeling and reanalyzing when the many design changes have to be evaluated. Details of the internal report are disclosed of commercial reasons.

stage, therefore *there is no doubt that integrated dynamic models will save time in (the consultants') daily work.* These numbers need to be verified by other consulting companies to be validated. At present time few consultants in the Danish building industry make use of integrated dynamic models, therefore it is difficult to verify Grontmijs records. More comments on Grontmijs records are found in section 5.1.1.

The developments of integrated dynamic models may have been driven by the objectives of High-Performance Buildings, but as the outcome of such developments have created more cost-efficient modeling and evaluation techniques these models can be used for other (financial) gains. As discussed in Section 4.5.3 the embedded knowledge in the models as a fortunate side effect of the Consequence based design approach, consultant engineers will be able to create even faster and more cost-efficient analyses with integrated dynamic models in the future. But there is no guarantee that the economic gains will benefit the creation of High-Performance Buildings.

4.5.5 Comments on optimization

"You can formulate the design problem as one of enumerating feasible solutions for consideration, or you can specify an objective function and search for optimal or good sub-optimal solutions." (Mitchell, 1998).

Basically both approaches to optimization are valid; however when it comes to applying optimization algorithms in the design process the latter approach is necessary. In terms of the definition of "optimal or good sub optimal solutions" challenges arise. The case study Good Year (Section 4.1.5) showed the option to include architectural (qualitative) considerations as constraints in the objective function. The main problem with this approach was the post processing needed to find a solution that was "optimal or a good sub optimal solution" in terms of the many architectural criteria. Even though Pareto ranking was shown to help in the process to navigate in the many good sub optimal solutions, there was and still is a problem with the process "of choosing a solution" in the option space. If optimization algorithms are to be used as showed in Good Year, it has to be very clear for all the involved parties in the design team that the "optimal" solutions are suggestions (not resulting solutions) to inform the design team. The solutions will always be lacking quality in some aspect of building design, partially because the objective function does not consider all aspects of design, and partially because the solutions are representations of the trade-offs between objectives that have been predefined.

The proposed method of utilizing agents (the Moth case, see Section 4.4.5) for in-state optimization was shown to work in a constructed case, however it has yet to be seen used in practice.

Therefore, integrated dynamic models with optimization are still far from ideal in their current form. High risks of excluding essential parts of building design evaluations (specifically the qualitative criteria) still persist.

5. Conclusion

The primary aim of the project was to envision, implement and document the use of integrated dynamic models. This thesis demonstrates ways that integrated dynamic models can be used in practice. The thesis demonstrates how integrated dynamic models may include building performance feedbacks, specifically feedbacks regarding energy consumption and indoor environment in the aim to create High-Performance Buildings. It further demonstrates the inclusion of quality defined performances un-associated with High-Performance Buildings. The thesis has discussed ways integrated dynamic models affect the design process and collaboration between building designers and simulationists. This was done by applying the approach of Consequence based design to five case studies, followed by documentation based on interviews, surveys and project related documentations derived from internal reports and similar sources.
The case studies include a mixed use building, office buildings of various complexities, a building renovation of housings and a hospital project. The Consequence based design approach was applied to all case studies in a way where integrated dynamic models have been used to inform the design team on building performance associated with the objectives of High-Performance Buildings.

The secondary aim was to envision, implement and document optimization processes when combined with integrated dynamic models. This thesis has investigated and demonstrated multi-objective optimization methods, including evolutionary algorithms and agent based modelling. Multi criteria optimization has been applied to one of the five case studies; it was followed up by several developments to improve optimization of High-Performance Buildings in the early design stage.
The Developments cover new software tools, methods to clarify objectives and clarifications of other elements in the early design stage, developments of new algorithms (optimization algorithm, ontology algorithms, and prediction algorithms) and developments of new calculation methods for building energy consumption and thermal indoor environment.

Visualization and facilitation of performance consequences

The ability to visualize and facilitate building performance with integrated dynamic models has been observed and shown to increase the (simulated) building performance. The results of the case studies suggest that integrated dynamic models are highly efficient in transforming quantitative performance metrics into visual feedback for building designers and other practitioners. The use of integrated dynamic models has been shown to help the design team to find the balance between what is possible, what is lawfully required and what design variables are most cost-efficient to change in terms of High-Performance Building objectives. The models help to communicate changes in a non-technical and easily understandable way, which was found highly valuable to all members of

the design team. For the five case studies performed, it can be concluded that the Consequence based design approach has shown to be efficient in clarifying performance based objectives in the design process.
Since there have been no direct ways to measure if the increased consequence feedback can improve building performance and increase the likelihood of High-Performance Buildings, surveys and interviews have been used as documentation. The results of the surveys indicate that the inclusion of integrated dynamic models is likely to increase building performance compared to traditional design approaches. As the surveys are based on limited data and a small number of projects more research is needed to fully conclude that integrated dynamic models and Consequence based design is increasing the building performance of buildings.

Building performance – one of many important criteria

When both qualitative and quantitative performance requirements are taken into consideration the design teams in several case studies have shown more interest in using integrated dynamic models. Thus, the success of creating High-Performance Buildings with integrated dynamic models is tied to the ability to include more than predefined objectives of High-Performance Buildings. With integrated dynamic models in the early design stage the simulationists have been able to simulate and evaluate more solutions (several magnitudes) than previously possible. Based on the surveys these changes to the early design removes a major risk for the simulationists in the subsequent design phases. In practice this has been hard to detect since none of the projects have been monitored during the later stages. Therefore whether or not Consequence based design and the extensive use of integrated dynamic models leads to a reduced number of changes to the building design in the later project stages still need further investigation.
The integrated dynamic models have shown ways to include architectural concerns not associated with High-Performance Buildings, and the ability to include qualitative (high visual quality) feedbacks and qualitative assessments.

Building performance optimization in the early design stages

The approach of Consequence based design has demonstrated the use of integrated dynamic models with optimization relating to High-Performance Building objectives. Optimization was found difficult to put into practice as the current optimization algorithms and methods have problems in handling both the qualitative and the quantitative objectives. Nevertheless, when optimization methods were implemented in the model, the results were used by the design team as any other feedback from an integrated dynamic model. The main differences are that integrated dynamic models with optimization *generate* solutions (thousands of solutions), and therefore post processing of solutions is necessary. The generation process and post process are biased towards the predefined performance objectives, and this make optimization with qualitative criteria difficult to handle. The integrated dynamic models with optimization, however, were found beneficial over existing optimization approaches, as qualitatively defined constraint functions with little effort could be implemented by the design team. The inclusion of constraint functions limited the search space into a finite and manageable space of interest. This reduced the number of simulations made by the coupled BPS tools and made the optimization viable in the early design stage. The integrated dynamic models with optimization have shown promise to help the design team in locating High-Performance Building designs that do include a narrow amount of quality defined constraint functions. Thus it can be concluded that optimization as a method can be included in integrated dynamic models, however the nature of optimization as a generational process still pose challenges in terms of artistic and qualitative control.

5.1 Research contributions

This Ph.D. project is an industrial Ph.D. which means that the research had the opportunity to be applied in practice.

5.1.1 Contribution to industry

Batteriet

The integrated dynamic models have been found resourceful over existing BPS tool models that were either too slow or required a large amount of knowledge and expertise to operate. The integrated dynamic models allowed an easy way to create a quasi-steady-state calculation method directly in the VPL. And it allowed dynamic couplings to spreadsheets (Excel). Thereby creating a new way to utilize existing (Grontmij's own) calculation methods such as a heat-balance equation and apply this directly on the architectural concept models. The integrated dynamic model was found more precise and faster than the traditional rules-of-thumb and design guidelines. The models showed the benefits to utilize automation in modelling. This created a freedom to sort, process, bundle and extract data for desired purposes. Options and processes that usually require large number of worker hours are with integrated dynamic models fully automated.

Kirk

From the Kirk project, it was established that the traditional evaluation approach of the energy consumption (utilizing the BPS tool Be10) in the very complex building geometry had led to under-estimation due to modelling and data exchange errors. When the subsequently integrated dynamic model was used with the plugin Termite (coupled Be10 solver) the following was found:

- A more valid basis for energy simulations can be established
- Integrated dynamic models create a whole range of new design variable analyses
- The building designers receive consequence feedback in a visual form that represents the exact model
- Integrated dynamic models have shown to facilitate evaluations that traditional BPS tools cannot

Nordfløjen

The integrated dynamic models used in the Nordfløjen project ensured a highly responsive performance based design process. The Consequence based design approach produced more results and was several magnitudes faster than traditional methods. The separated (but dynamically integrated) tools ensured quality assurance, as the simulationists operated the BPS input/output and the building designers handled the geometry model in their own design tools.

Grontmij considered Nordfløjen as one of the great successes in newer history in terms of integrated design. Daylight conditions and energy consumption were optimized by controlling the size of individual windows of a 55,000 m^2 building façade. This created a High-Performance hospital which was found much cheaper in terms of building operation than the client required. A task, which according Grontmij, *in no way could have been done* without integrated dynamic models and implementation of Be10 (Termite). The project also showed the benefit of the Consequence based design approach as the daylight and energy criteria were successfully included in synergy with the architects' criteria.
The parametric abilities of the model made it possible to showcase numerous variations of the same concept, thereby frontloading responds to changes quickly and visually.

Nordic Built

The Nordic Built project has shown that BPS tools in some cases are incapable of qualifying the consequence of design choices in the early design stage, even when clear objectives have been defined. This especially applies to social sustainability. The integrated dynamic models demonstrated ways to clarify and visualize performance evaluations based on qualitative evaluations. The use of qualitative evaluation methods was found to fit well with the objectives of Consequence based design, especially in terms of difficult assessments on multidisciplinary problems such as social and environmental building renovation projects.

Good Year

The integrated dynamic models used in the Good Year project showed promise in multi criteria optimization. Applying optimization (such as using evolutionary algorithms like SPEA-II) to the model, automating the generation of variations and comparing alternatives in short time was difficult to do without dedicated optimization tools.
The integrated dynamic models with optimization allowed the design team to focus on very specific design problems that include qualitative criteria defined as constraint functions. This had not been possible without the integrated dynamic model.

Commercial perspectives

The many ways integrated dynamic models have been used have demonstrated integrated dynamic models to be very flexible. Grontmij has identified the models to be most effectively used in all large and medium-sized projects where the design team needs to evaluate the building relating to minimum requirements e.g. for daylight and energy consumption. Some types of evaluations such as quantitative shadow-analyses (Nordfløjen) are not possible without integrated dynamic models. Grontmij has in an internal business report estimated that the use of integrated dynamic models can save the company 25% of the company's resources (mainly referring to the saved working hours and only in this particular design stage) if used in the early design stage. This estimation, is based on feedback from a limited amount of projects and based on Grontmijs own experiences. Similar conclusions are needed to be made from other consultancies using integrated dynamic models, for this to be verified. Worth noticing is that Grontmij's internal report state that the savings will likely to be found in future projects if the company decide to enter early design projects, but the report also suggest that the resources saved *should be put into making more and more detailed analyses to further strengthen the building performance of the particular project.* Therefore the large amount of savings in working hours may be questionable. After all since consulting companies are competing on competency *and* price the saved working hours may partly or fully be invested in improving the consultancy quality. The interesting notion though, is the "saved" working hours could substitute "better building performance". At these points there have been little evidence in this claim and more research is needed to state if this is true in general sense. Also whether these savings apply to the entire design team has not been determined. As an advise to consulting companies seeking to utilize integrated dynamic models, it must be held that even if there are two large potentials in these types of models; the potential of better collaboration between the building designer and the simulationist, and the potential of more efficient use of simulation tools, these potentials cannot be separated. This can also be said in another less elegant way, to exploit the efficiency of integrated dynamic models to save time, the simulationist must reach out to the building designer and collaborate from the earliest possible stage. In the end it all comes down to how people relate on personal and professional levels and no model can replace that.

5.1.2 Contribution to academia

13 student projects have been co-supervised or subject to external assistance during the Ph.D. project. 5 articles (Negendahl and Nielsen, 2015; Negendahl, 2015a, 2014b; Negendahl et al., 2015, 2014) and one poster presentation have been produced (Negendahl, 2014a). The author has been interviewed or contributed otherwise with material to the following articles: (Gram, 2014; Mueller et al., 2013; Peters, 2012). One workshop on Termite was organised by the author (*Energi og Parametri* Glostrup Denmark, May 2014), and the author has participated in two external workshops (*SmartGeometry* New York 2012, *DCEE3* Copenhagen 2014). The author has presented various findings relating to

Consequence based design at four conferences (*eCAADe* Newcastle 2014, *DCEE3* Kgs. Lyngby 2014, *iiESI* Copenhagen 2014, *BIPS* Nyborg 2012).

Developments

Integrated dynamic models were found to be a very strong and versatile platform for prototyping new tools, new methods, new algorithms and new concepts. The sentient integrated dynamic model (section 4.4.6) is one example of how a student may implement a concept of *user preference listener functions*. Other students under supervision during this project have prototyped several other concepts or algorithms that have not been mentioned here, but have been interesting and inspiring to follow. If nothing else, the integrated dynamic model as a research and test platform is deemed to generate many innovative ideas. And hopefully some of these prototypes may propagate into practice and bring momentum to the Consequence based design approach.

Prototyping also applies to the development of new tools and algorithms, such as Termite, HQSS and Moth. These developments have been beneficial for many of the case studies. For example, Termite has provided new possibilities for the use of BPS tool Be10. The different developments share the following benefits:

- Fast and parallel simulations/evaluations
- Linking and combining different types of simulations/evaluations with other simulations/evaluations
- The ability to apply optimization using the VPL as a middleware to different types of optimization algorithms
- Visualization of consequences derived from BPS results
- Visualization of numerical analyses unassociated with BPS results

These benefits are the direct result of integrated dynamic models. Therefore, new developments of tools, algorithms and methods prototyped with integrated dynamic models will be subject to similar benefits.

5.2 Future work

Inclusion of stakeholders

Future works where practitioners utilize integrated dynamic models in the aim to create High-Performance Buildings are a ratification of the Consequence based design approach. Practitioners in all disciplines of building design might be able to benefit from integrated dynamic models. However, at this point only the building designer and the simulationist have been included in the design process. Therefore, more applications of Consequence based design are needed to determine if the inclusion of e.g. the contractor, the client, and the end user may increase the chance of creating High-Performance Buildings.

High-Performance Building objectives

The main focus of this project has been to improve the energy consumption of buildings and the thermal indoor environment. High-Performance Buildings concern many more building performances that may be derived from BPS tools. Therefore, more research and development in the coupling of BPS tools and VPLs is needed to fully cover and connect all performance of High-Performance Buildings.

Optimization

More research is needed to fully integrate qualitative assessments in optimization. The work on agent based modelling for optimization has shown promise to change and adjust the optimisation process based on continuous qualitative assessment. However, more research is needed to improve agent based optimization algorithms and how these algorithms can be applied in practice while maintaining quality assurances.

Design process

Consequence based design has been confined to the early design stage as changes are less expensive and the relative impact is higher. Nevertheless, the transition from the early design stages into the construction stages and even further has not been addressed. Using integrated dynamic models in later design stages may be beneficial, but to determine this more research and applications in practice are needed. It is believed by the author that more conventional collaboration BIM models (such as IFC-models) are more beneficial for the later design stages. Therefore, more research in combining integrated dynamic models and e.g. IFC is needed.

References

Ahsrae, 2005. Energy estimating and modelling methods, models and approach. ASHRAE Handb. - Fundam. 39.

Amor, R., Faraj, I., 2001. Misconceptions about integrated project databases. J. Inf. Technol. Constr. 6, 57–68.

Andresen, I., 2000. A Multi-Criteria Decision-Making Method for Solar Building Design. Norwegian University of Science and Technology.

Attia, S., 2012. A Tool for Design Decision Making. Université catholique de louvain.

Attia, S., Hensen, J.L.M., Beltrán, L., De Herde, A., 2012. Selection criteria for building performance simulation tools: contrasting architects' and engineers' needs. J. Build. Perform. Simul. 5, 155–169.

Augenbroe, G., 1992. Integrated building performance evaluation in the early design stages. Build. Environ. 27, 149–161.

Augenbroe, G., 2001. Building Simulation Trends going into the new millenium. In: Seventh International IBPSA Conference. Rio de Janeiro, Brazil, pp. 15–28.

Augenbroe, G., 2002. Trends in building simulation. Build. Environ. 37, 891–902.

Autodesk, 2013a. Dynamo.

Autodesk, 2013b. Ecotect.

Autodesk, 2013c. Revit.

Autodesk, 2013d. Vasari.

Balcomb, D., Andresen, I. Hestnes, A.G., Aggerholm, 2002. MultiCriteria Decision Making (MCDM-23) - A Method for Specifying and Prioritising Criteria and Goals in Design, IEA SHC Task 23 Optimization of Solar Energy Use in Large Buildings.

Bambardekar, S., Poerschke, U., 2009. The architect as performer of energy simulation in the early design stage. In: Proceedings of the 11 Th International IBPSA Conference. Glasgow, Scotland. pp. 1306–1313.

Banke, T., 2013. Parametri i praksis - Generativ performance i arkitektur. Det Kongelige Danske Kunstakademis Skoler for Arkitektur, Design og Konservering.

Barbati, M., Bruno, G., Genovese, a., 2012. Applications of agent-based models for optimization problems: A literature review. Expert Syst. Appl. 39, 6020–6028.

Bazjanac, V., 2004. Building energy performance simulation as part of interoperable software environments. Build. Environ. 39, 879–883.

Bazjanac, V., Kiviniemi, A., 2007. Reduction, simplification, translation and interpretation in the exchange of model data. In: Proceedings of the 24th Conference on Bringing ITC Knowledge to Work. Moribar, Slovenia, pp. 163–168.

Beck, K., 1999. Extreme Programming Explained: Embrace Change, XP Series.

Benedikt, M., 1979. To take hold of space: isovists and isovist fields. Environ. Plan. B 6, 47–65.

Bentley, 2013. GenerativeComponents.

Bentley, 2014. MicroStation V8i.

Bento, J., Feijó, B., 1997. An agent-based paradigm for building intelligent CAD systems. Artif. Intell. Eng. 11, 231–244.

Bergek, A., Jacobsson, S., Carlsson, B., Lindmark, S., Rickne, A., 2008. Analyzing the functional dynamics of technological innovation systems: A scheme of analysis. Res. Policy 37, 407–429.

Bernal, M., Eastman, C., 2011. Top-down Approach to Embed Design Expertise in Parametric Objects for the Automatic Generation of a Building Service Core. CAAD Futur. 2011 149–164.

Beth Tubbs, P.E., 2001. CIB TG37 Performance Based Building Regulatory Systems. Northbridge.

Bienert, S., Schutzenhofer, C., Leopoldsberger, G., Bobsin, K., Leutgob, K., Huttler, W., Popescu, D., Mladin, E.-C., Koch, D., Edvardsen, D.F., 2012. Methodologies for Integration of Energy Performance and Life-Cycle Costing Indicators into Property Valuation Practice, www.immovalue.org. KPMG Financial Advisory Services GmbH.

Biloria, N., 2011. InfoMatters, a multi-agent systems approach for generating performative architectural formations. Int. J. Archit. Comput. 09, 205–222.

Bleil de Souza, C., 2012. Contrasting paradigms of design thinking: The building thermal simulation tool user vs. the building designer. Autom. Constr. 22, 112–122.

Boehm, B.W., 1984. Software Engineering Economics, IEEE Transactions on Software Engineering.

Borden, R.J., 2008. A Brief History of SHE : Reflections on the Founding and First Twenty Five Years of the Society for Human Ecology. Hum. Ecol. Forum 15, 95–108.

Bradley, D., Kummert, M., 2005. New Evolutions in Trnsys – a Selection of Version 16 Features. Ninth Int. IBPSA Conf. 107–114.

Brooks, J.F.P., 2010. The Design of Design: Essays from a Computer Scientist.

BSI, 15251, E.S., 2006. prEN 15251, Indoor environmental input parameters for design and assessment of energy performance of buildings- addressing indoor air quality , thermal environment , lighting and acoustics.

Buelow, P. Von, Falk, A., Turrin, M., 2010. Optimization of structural form using a genetic algorithm to search associative parametric geometry. Struct. Archit. 195–196.

BuildingSMART, 2013. IFC - BuildingSMART [WWW Document]. URL http://www.buildingsmart.org/about/operating-vision/core-programs/

BuildingSmart.org, 2013. BuildingSMART [WWW Document]. URL BuildingSmart.org

Burhenne, S., Jacob, D., Henze, G.P., 2011. Sampling based on sobol sequences for monte carlo techniques applied to building simulations. In: 12th Conference of International Building Performance Simulation Association. pp. 14–16.

Burry, M., 2013. Scripting cultures: Architectural design and programming. AD Primers.

Byrne, J., Fenton, M., Hemberg, E., Mcdermott, J., Neill, M.O., Shotton, E., Nally, C., 2011a. Combining Structural Analysis and Multi-Objective Criteria for Evolutionary Architectural Design. Archit. Des. 204–213.

Byrne, J., Hemberg, E., O'Neill, M., 2011b. Interactive operators for evolutionary architectural design. In: Proceedings of the 13th Annual Conference Companion on Genetic and Evolutionary Computation - GECCO '11. p. 43.

Caldas, L., 2008. Generation of energy-efficient architecture solutions applying GENE_ARCH: An evolution-based generative design system. Adv. Eng. Informatics 22, 59–70.

Caldas, L., Norford, L., 2001. Architectural Constraints in a Generative Design System: interpreting energy consumption levels. In: Seventh International IBPSA Conference. pp. 13–15.

Caldas, L.G., 2001. An Evolution-Based Generative Design System: Using Adaption to Shape

Architectural Form. Massachusetts Institute of Technology.

Christensen, C., Horowitz, S., Givler, T., Courtney, A., Barker, G., 2005. BEopt : Software for Identifying Optimal Building Designs on the Path to Zero Net Energy Preprint. In: ISES 2005 Solar World Congress.

Christensen, J.E., Schiønning, P., Dethlefsen, E., 2013. Comparison of Simplified and Advanced Building Simulation Tool With Measured Data. In: 13th Conference of Inernational Building Performance Simulation Association. Chambéry, France, pp. 2357–2364.

Citherlet, S., Clarke, J. a., Hand, J., 2001. Integration in building physics simulation. Energy Build. 33, 451–461.

Clarke, J.A., 2001. Energy Simulation in Building Design. In: Energy Simulation in Building Design. Taylor & Francis, pp. 1–20.

Corgnati, S., Bednar, T., Jang, Y., Yoshino, H., Filippi, M., Danov, S., Djuric, N., Schweiker, M., Ghiaus, C., Capozzoli, A., Fabi, V., 2013. Statistical analysis and prediction methods Separate Document Volume V.

Courtney L. Fromberg, Situ, Y., Issaa, R., 2015. Explorations, Challenges, + Possible Future Solutions. In: Computational Design Progression, and Future Generation.

Danish Building Regulations, 2013. Lavenergiramme for kontorer, skoler, institution m.m. Section 7.2.4.2. http://bygningsreglementet.dk/br10_03_id161/0/42 2013.

Danish Energy Agency, 2012. Energy Efficiency Policies and Measures in Denmark 2012, Monitoring of Energy Efficiency in EU 27, Denmark (ODYSSEE-MURE).

Davidsson, P., Persson, J. a, Holmgren, J., 2007. On the Integration of Agent-Based and Mathematical Optimization Techniques. Proc. 1st KES Int. Symp. Agent Multi-Agent Syst. Technol. Appl. 1–10.

Davis, D., 2013a. Modelled on Software Engineering: Flexible Parametric Models in the Practice of Architecture. RMIT University.

Davis, D., 2013b. Yeti3d.

Davis, D., Peters, B., 2013. Design Ecosystems: Customising the Architectural Design Environment with Software Plug-ins. Archit. Des. 83, 124–131.

Davis, D., Villaggi, L., Bailey, C., 2014. Not Everything is Captured by the Fitness Function. Colon 4 2.

Daysim.ning.com, 2013. Daysim.

de Souza, C.B., 2009. A critical and theoretical analysis of current proposals for integrating building thermal simulation tools into the building design process. J. Build. Perform. Simul. 2, 283–297.

Deru, M., Torcellini, P., 2005. Performance Metrics Research Project – Technical Report.

Dirkse, S.P., Ferris, M.C., 1996. A Pathsearch damped Newton method for computing general equilibria. Ann. Oper. Res. 68, 211–232.

Dogan, T., 2014. ArchSim Energy Modeling [WWW Document]. http://www.food4rhino.com/project/archsim-em.

Dong, B., Cao, C., Lee, S.E., 2005. Applying support vector machines to predict building energy consumption in tropical region. Energy Build. 37, 545–553.

Duarte, J.P., 2001. Customizing mass housing : a discursive grammar for Siza's Malagueira houses. Massachusetts Institute of Technology.

Eastman, C.M., Teicholz, P., Sacks, R., Liston, K., 2011. BIM Handbook: A Guide to Building Information Modeling for Owners, Managers, Architects, Engineers, Contractors, and

Fabricators. Wiley.

Eberhart, Yuhui, S., 2001. Particle swarm optimization: developments, applications and resources. Evol. Comput. 2001. Proc. 2001 Congr.

Edwards, L., Torcellini, P., 2002. A Literature Review of the Effects of Natural Light on Building Occupants. Contract 58.

Ehrich, A.B., Haymaker, J.R., 2011. Multiattribute interaction design: An integrated conceptual design process for modeling interactions and maximizing value. Artif. Intell. Eng. Des. Anal. Manuf. 26, 85–101.

Eisenhower, B., O'Neill, Z., Narayanan, S., Fonoberov, V. a., Mezić, I., 2012. A methodology for meta-model based optimization in building energy models. Energy Build. 47, 292–301.

Ellis, P.G., Griffith, B., Long, N., Torcellini, P., 2006. Automated Multivariate Optimization Tool for Energy Analysis Preprint. Energy Build.

EPA, 1991. Indoor Air Facts No. 4 Sick building syndrome, Air and Radiation.

Equa, 2014. IDA Indoor Climate and Energy IDA Indoor Climate and Energy simulation software [WWW Document]. URL http://www.equa-solutions.co.uk/en/software/idaice

EU, 2010. Directive 2010/31/EU of the European Parliament and of the Council of 19 May 2010 on the energy performance of buildings (recast). Off. J. Eur. Union 13–35.

EU, 2011. JESSICA implementation in the EU Member States. State of play.

European Investment Bank, 2010. The European Investment Bank ELENA – European Local Energy Assistance.

European Parliament, 2009. Directive 2009/28/EC of the European Parliament and of the Council of 23 April 2009. Off. J. Eur. Union 140, 16–62.

Fanger, P.O., 1973. Assessment of thermal comfort practice. Occup. Environ. Med. 30, 313–324.

Feng, K., Jensen, Z., Zhang, C., Grunewald, J., Nicolai, A., Fechner, H., 2013. CHAMPS MultiZone bulding application [WWW Document]. URL http://champs.syr.edu/software/champs_mz/champs_mz.html

Fialho, Á., Hamadi, Y., Schoenauer, M., 2011. Optimizing architectural and structural aspects of buildings towards higher energy efficiency. In: Proceedings of the 13th Annual Conference Companion on Genetic and Evolutionary Computation - GECCO '11. p. 727.

Frayling, C., 1993. Research in Art and Design. R. Coll. Art Res. Pap.

Gaspar-Cunha, A., Loyens, D., van Hattum, F., 2011. Aesthetic design using multi-objective evolutionary algorithms. In: Evolutionary Multi-Criterion Optimization. Springer, pp. 374–388.

gbXML.org, 2014. Open Green Building XML gbXML [WWW Document]. URL http://www.gbxml.org/ (accessed 9.1.12).

Gehry Technologies, 2013. Digital Project.

Georgescu, M., Eccles, E., Manjunath, V., Swindle, E., Mezi´, I., 2010. Machine Learning Methods for site-level building energy forecasting and data rectification.

Gerber, D.J., Lin, S.E., 2012. Synthesizing design performance : an evolutionary approach to multidiciplinary. In: ACADIA 2012 Synthetic Digital Ecologies. pp. 66–75.

Gram, D., 2014. Parametrisk designproces sikrer højere kvalitet i byggeriet. LYS-03 4–6.

Granholm, L., 2012. BIM at Tekla Tekla Corporation. In: ECPPM Conference 2012.

Greenberg, D., Pratt, K., Hencey, B., Jones, N., Schumann, L., Dobbs, J., Dong, Z., Bosworth, D., Walter, B., 2013. Sustain: An experimental test bed for building energy simulation.

Energy Build. 58, 44–57.

Grierson, D.E., 2008. Pareto multi-criteria decision making. Adv. Eng. Informatics 22, 371–384.

Gupta, A., 2013. Developing a BIM-based methodology to support renewable energy assessment of buildings. Cardiff University.

Gupta, A., Cemesova, A., Hopfe, C., Rezgui, Y., Sweet, T., 2013. Development of a Solar PV Simulation Tool Compatible with Industry Foundation Classes. In: 13th Conference of International Building Performance Simulation Association. 13th Conference of International Building Performance Simulation Association, Chambery.

Harding, J., Joyce, S., Shepherd, P., Williams, C., 2012. Thinking Topologically at Early Stage Parametric Design. In: Advances in Architectural Geometry 2012. Springer Vienna, pp. 67–76.

Hensen, J., 2002. Simulation for performance based building and systems design: some issues and solution directions. In: 6th Conf on Design and Decision Support System in Architecture and Urban Planning. pp. 186–199.

Hensen, J.L.M., 2004. Towards more effective use of building performance simulation in design. Proc. 7th Int. Conf. Des. Decis. Support Syst. Arch. Urban Planin. 2–5.

Hermund, A., 2009a. Building information modeling in the architectural design phases: And why compulsory BIM can provoke distress among architects. eCAADe 1–8.

Hermund, A., 2009b. Building Information Modeling in the Architectural Design Phases And Why Compulsory BIM can Provoke Distress Among Architects. eCAADe 1–8.

Hermund, A., 2012. Anvendt 3D modellering og parametrisk formgivning. Design. Det Kongelige Danske Kunstakademis Skoler for Arkitektur, Design og Konservering.

Holzer, D., 2010. Optioneering in Collaborative Design Practice. Int. J. Archit. Comput. 8, 165–182.

Holzer, D., Downing, S., Tengono, Y., 2009. Delivering digital architecture in australia, DDAA Presentation, Australian Research Council.

Hooke, R., Jeeves, T. a., 1961. "Direct Search'' Solution of Numerical and Statistical Problems. J. ACM 8, 212–229.

Hopfe, C., Hensen, J., 2009. Experiences testing enhanced building performance simulation prototypes on potential user group. 11th IBPSA Build. Simul. Conf. 1976–1982.

Hopfe, C., Struck, C., Harputlugil, G., Hensen, J., Wilde, P. De, 2005. Exploration of the use of building performance simulation for conceptual design. In: Proceedings of IBPSA-NVL Conference. pp. 1–8.

Hopfe, C.J., 2009. Uncertainty and sensitivity analysis in building performance simulation for decision support and design optimization. Tech. Univ. Eindhoven. Technische Universiteit Eindhoven.

Hopfe, C.J., Emmerich, M.T.M., Marijt, R., Hensen, J., 2012. Robust multi-criteria design optimization in Building design. In: Proceedings of the 1st IBPSA-England Builiding Performance Simulation Association, BSO12. pp. 118–125.

Hopfe, C.J., Hensen, J.L.M., 2011. Uncertainty analysis in building performance simulation for design support. Energy Build. 43, 2798–2805.

http://www.breeam.org/. BREEAM [WWW Document]. Green Build. Counc. UK. URL http://www.breeam.org/ (accessed 1.1.12).

Http://www.dgnb.de/. DGNB [WWW Document]. Ger. Sustain. Build. Counc. URL http://www.dgnb.de/ (accessed 1.1.12).

Http://www.dk-gbc.dk/. DGNB (DK) [WWW Document]. Green Build. Counc. Denmark. URL http://www.dk-gbc.dk/ (accessed 1.1.12).

http://www.regionh.dk/, 2011. Fremtidens Rigshospital nu på tegnebrættet 7. december 2011.

http://www.rigshospitalet.dk/, 2010. Idéoplæg, Righsospitalets byggeprojekt 9.marts 2010.

http://www.usgbc.org/leed. LEED [WWW Document]. Green Build. Counc. USA. URL http://www.usgbc.org/leed (accessed 1.1.12).

Innovationsfonden, 2015. Innovationsfonden [WWW Document]. URL http://innovationsfonden.dk/en/investment/industrial-researcher (accessed 3.27.15).

Integrated Environmental Solutions, 2013. IESVE.

ISO, 2008. EN ISO 13790:2008 Energy performance of buildings - Calculation of energy use for space heating and cooling.

Jakubiec, J.A., Reinhart, C.F., 2011. DIVA 2.0 : Integrating daylight and thermal simulations using Rhinosoros 3D , DAYSIM and ENERGYPLUS. In: 12th Conference of International Building Performance Simulation Association. pp. 14–16.

James J. Hirsch & Associates, 2013. DOE-2.

Kalay, Y.E., 1999. Performance-based design. Autom. Constr. 8, 395–409.

Kalay, Y.E., 2001. Enhancing multi-disciplinary collaboration through semantically rich representation. Autom. Constr. 10, 741–755.

Kalogirou, S. a., 2000. Applications of artificial neural-networks for energy systems. Appl. Energy 67, 17–35.

Kanters, J., Horvat, M., Dubois, M.C., 2014. Tools and methods used by architects for solar design. Energy Build. 68, 721–731.

Karlshøj, J., See, R., Davis, D., 2012. An Integrated Process for Delivering IFC Based Data Exchange.

Kataras, P., 2010. Multi-objective Performance Optimization Of Building Form based on the principle of analytical geomertry. Architecture. University College London.

Kennedy, J., Eberhart, R., 1995. Particle swarm optimization. Neural Networks, 1995. Proceedings., IEEE Int. Conf. 4, 1942–1948 vol.4.

Kiesler, F., 1939. On Correalism and Biotechnique. Archit. Rec. 60–66.

Klitgaard, J., Kirkegaard, P.H., Mullins, M., 2006. On the Integration of Digital Design and Analysis Tools. In: Digital Architecture and Construction. pp. 187–196.

Kowalski, R., 1979. Algorithm = logic + control. Commun. ACM 22, 424–436.

Lam, K.., Wong, N.., Mahdavi, a, Chan, K.., Kang, Z., Gupta, S., 2004. SEMPER-II: an internet-based multi-domain building performance simulation environment for early design support. Autom. Constr. 13, 651–663.

Lauridsen, P.K.B., Petersen, S., 2014. Integrating Indoor Climate , Daylight and Energy Simulations in Parametric Models and Performance-Based Design. In: Third International Workshop on Design in Civil and Environmental Engineering.

Laustsen, J., Ruyssevelt, P., Staniaszek, D., Zinetti, S., Strong, D., 2011. Europe ' s buildings under the microscope. Buildings Performance Institute Europe.

LBNL, 2014. Simergy [WWW Document]. Lawrence Berkeley Natl. Lab. URL http://simergy.lbl.gov/

Lidwell, W., Holden, K., Butler, J., 2003. Universal Principles of Design.

Limbourg, P., Kochs, H.-D., 2008. Multi-objective optimization of generalized reliability design problems using feature models—A concept for early design stages. Reliab. Eng. Syst. Saf. 93, 815–828.

Lin, S., Gerber, D.J., 2014. Evolutionary energy performance feedback for design : Multidisciplinary design optimization and performance boundaries for design decision support. Energy Build. 84, 426–441.

Lin, S.E., Gerber, D.J., Hall, W., Angeles, L., 2013. Designing - In Performance : Towards Cloud Based Simulation and Multidisciplinary Design Solution Space Search. In: SimAUD 2013 Symposium on Simulation for Architecture and Urban Design.

Liu, B., Wang, L., Jin, Y.-H., Tang, F., Huang, D.-X., 2005. Improved particle swarm optimization combined with chaos. Chaos, Solitons & Fractals 25, 1261–1271.

Liu, K., Hsueh, S., Wu, W., Chen, Y., 2012. A DFuzzy-DAHP Decision-Making Model for Evaluating Energy-Saving Design Strategies for Residential Buildings. Energies 5, 4462–4480.

Löhnert, G., Dalkowski, A., Sutter, W., 2003. Integrated Design Process Task 23 Subtask B Design Process Guidelines.

Machairas, V., Tsangrassoulis, A., Axarli, K., 2014. Algorithms for optimization of building design : A review, Renewable and Sustainable Energy Reviews 31, 101–112.

Machwe, a. T., Parmee, I.C., 2007. Multi-objective analysis of a component-based representation within an interactive evolutionary design system. Eng. Optim. 39, 591–613.

MacLeamy, P., 2013. Bim-Bam-Boom! The future of the building industry. HOK, Youtube.

Mahdavi, A., 1998. Computational decision support and the building delivery process: a necessary dialogue. Autom. Constr. 7, 205–211.

Mahdavi, A., 2004. Reflections on computational building models. Build. Environ. 39, 913–925.

Mahdavi, A., Mathew, P., Kumar, S., Wong, N.-H., 1997. Bi-directional computational design support in the SEMPER environment. Autom. Constr.

Malkawi, A.M., 2004. Developments in environmental performance simulation. Autom. Constr. 13, 437–445.

Mallory-Hill, S.M., 2004. Supporting Strategic Design of Workplace Environments with Case-Based Reasoning. Technical University of Eindhoven.

Martin Tamke, Stasiuk, D., Thomsen, M.R., 2013. ALIVE: Designing with aggregate behaviour in self-aware systems. In: Gengnagel, C., Nembrini, J., Kilian, A., Scheurer, F. (Eds.), Rethinking Prototyping : Proceedings of the Design Modelling Symposium. Universität der Künste Berlin, Berlin, p. 18.

Mcelroy, L.B., Clarke, J. a, 1999. Embedding simulation within energy sector businesses. In: Proceedings of the 6th IBPSA Conference (Building Simulation 99). Kyoto, pp. 262–268.

Mela, K., Tiainen, T., Heinisuo, M., 2012. Advanced Engineering Informatics Comparative study of multiple criteria decision making methods for building design. Adv. Eng. Informatics 26, 716–726.

Michael Pelken, P., Zhang, J., Chen, Y., Rice, D.J., Meng, Z., Semahegn, S., Gu, L., Henderson, H., Feng, W., Ling, F., 2013. “Virtual Design Studio”—Part 1: Interdisciplinary design processes. Build. Simul. 6, 235–251.

Mitchell, W.J., 1975. Techniques of automated design in architecture: A survey and evaluation. Comput. Urban Soc. 1, 49–76.

Mitchell, W.J., 1977. Computer-Aided Architectural Design.

Mitchell, W.J., 1998. Articulate design of free-form structures. Artif. Intell. Struct. Eng. Lect. Notes Comput. Sci. Volume 145, 223–224.

Moosberger, S., 2009. Application of specific models in a general simulation tool. In: Building Simulation, Eleventh International IBPSA Conference. pp. 1603–1608.

Mora, R., Bédard, C., Rivard, H., 2008. A geometric modelling framework for conceptual structural design from early digital architectural models. Adv. Eng. Informatics 22, 254–270.

Mueller, V., Drury, B., Crawley, Zhou, X., 2013. Prototype Implementation of a Loosely Coupled Design Performance Optimisation Framework. In: Proceedings of the 18th International Conference on Computer-Aided Architectural Design Research in Asia (CAADRIA). Hong Kong, Singapore, pp. 675–684.

Nations, U., 1992. Status of Ratification of the Convention, United Nations Framework Convention on Climate Change.

Nederveen, S. Van, Bektas, E., Luiten, B., 2013. Distributed modeling for road authorities. In: Proceedings of the 30th CIB W78 International Conference. pp. 642–651.

Negendahl, K., 2014a. Parametric City Scale Energy Modeling Perspectives on using Termite in city scaled models. In: iiESI European Workshop. Kgs. Lyngby, Denmark.

Negendahl, K., 2014b. Parametric design and analysis framework with integrated dynamic models. In: Proceedings of the 3rd International Workshop on Design in Civil and Environmental Engineering. Kgs. Lyngby, Denmark.

Negendahl, K., 2014c. Termite. http://cobalab.dk/2014/07/15/termite-1-0/.

Negendahl, K., 2015a. Building Performance Simulation in the early design stage: An introduction to Integrated Dynamic Models. Autom. Constr. 54, 39–53.

Negendahl, K., 2015b. Building Performance Simulation in the early design stage: An introduction to Integrated Dynamic Models. Autom. Constr.

Negendahl, K., Nielsen, T.R., 2015. Building energy optimization in the early design stages: A simplified method. Energy Build. 105, 88–99.

Negendahl, K., Perkov, T., Heller, A., 2014. Approaching Sentient Building Performance Simulation Systems. In: Proceedings of eCAADe 2014. Newcastle, UK.

Negendahl, K., Perkov, T., Kolarik, J., 2015. Agent-based decision control - how to appreciate multivariate optimisation in architecture. In: 5th Design Modelling Symposium - Modelling Behaviour. Copenhagen, Denmark.

Nembrini, J., Samberger, S., Labelle, G., 2014. Parametric scripting for early design performance simulation. Energy Build. 68, 786–798.

Nguyen, A.-T., Reiter, S., Rigo, P., 2014. A review on simulation-based optimization methods applied to building performance analysis. Appl. Energy 113, 1043–1058.

Nielsen, T.R., 2002. Optimization of buildings with respect to energy and indoor environment. Technical University of Denmark.

Nielsen, T.R., 2005. Simple tool to evaluate energy demand and indoor environment in the early stages of building design. Sol. Energy 78, 73–83.

Nrel, 2013. Opt-E-Plus Software for Commercial Building Optimization. Natl. Renew. Energy Lab. U.S. DOE.

NREL U.S. Department of Energy, 2014. OpenStudio [WWW Document]. URL http://openstudio.nrel.gov/

Palonen, M., Hamdy, M., Hasan, A., 2013. MOBO A New Software for Multi-Objective Building Performance Optimization. In: 13th Conference of International Building Performance Simulation Association.

Parmee, I.C., Bonham, C.R., 2000. Towards the support of innovative conceptual design through interactive designer / evolutionary computing strategies. Artif. Intell. Eng. Des. Anal. Manuf. 14, 3–16.

Paulson Jr., B.C., 1976. Designing to Reduce Construction Costs. J. Constr. Div.

Pauwels, P., Van Deursen, D., De Roo, J., Van Ackere, T., De Meyer, R., Van de Walle, R., Van Campenhout, J., 2011. Three-dimensional information exchange over the semantic web for the domain of architecture, engineering, and construction. Artif. Intell. Eng. Des. Anal. Manuf. 25, 317–332.

Pedersen, F.Ø., 2006. A method for optimizing the performance of buildings. Technical University of Denmark.

Perez, R., Ineichen, P., Seals, R., Michalsky, J., Stewart, R., 1990. Modeling daylight availability and irradiance components from direct and global irradiance. Sol. Energy 44, 271–289.

Peters, T., 2012. Smartgeometry. Arkitekten 69–70.

Petersen, S., 2011. Simulation-based support for integrated design of new low-energy office buildings. Technical University of Denmark.

Petersen, S., Bryder, J., Levinsen, K., Strunge, J., 2014. Method for Integrating Simulation-Based Support in the Building Design Process. In: Proceedings of the 3rd International Workshop on Design in Civil and Environmental Engineering..

Petersen, S., Hviid, C.A., 2012. iDbuild.

Plume, J., Mitchell, J., 2007. Collaborative design using a shared IFC building model—Learning from experience. Autom. Constr. 16, 28–36.

Qian, Z., Seepersad, C., Joseph, V., 2006. Building surrogate models based on detailed and approximate simulations. J. of Mech. Design 128, 668-777.

R. Stouffs, P. Janssen, S. Roudavski, B.T., 2013. A design method for multicriteria optimization of low exergy architecture. In: CAADRIA 2013. pp. 117–126.

Rabl, a., Rialhe, a., 1992. Energy signature models for commercial buildings: test with measured data and interpretation. Energy Build. 19, 143–154.

Raphael, B., 2011. Multi-criteria decision making for collaborative design optimization of buildings. Built Environ. Proj. Asset Manag. 1, 122–136.

Reinhart, C.F., Andersen, M., 2006. Development and validation of a Radiance model for a translucent panel 38, 890–904.

Robert McNeel & Associates, 2013a. Rhino.

Robert McNeel & Associates, 2013b. Grasshopper3d.

Rose, C.M., Bazjanac, V., 2013. An algorithm to generate space boundaries for building energy simulation. Eng. Comput.

Roudsari, M., Mackey, C., Yezioro, A., Harriman, C.S., Chopson, P., Ahuja, S., 2014. Honeybee. https://github.com/mostaphaRoudsari/Honeybee.

Rutten, D., 2010. Evolutionary Principles applied to Problem Solving using Galapagos. In: AAG10. Vienna.

Salminen, M., Palonen, M., Sirén, K., 2012. Combined energy simulation and Multi-criteria optimisation of a LEED-certified building. In: Building Simulation and Optimization Conference. Loughborough, UK, pp. 372–377.

Sanguinetti, P., Abdelmohsen, S., Lee, J., Lee, J., Sheward, H., Eastman, C., 2012. General system architecture for BIM: An integrated approach for design and analysis. Adv. Eng. Informatics 26, 317–333.

Sanguinetti, P., Bernal, M., El-khaldi, M., Erwin, M., 2010. Real-Time Design Feedback : Coupling Performance-Knowledge with Design Iteration for Decision-making. Symp. Simul. Archit. Urban Des. 65–72.

Sanyal, J., Edwards, R., Ridge, O., Building, M.H.K., 2013. Supercomputer Assisted Generation of Machine Learning Agents for the Calibration of Building Energy Models. In: Proceedings of the Conference on Extreme Science and Engineering Discovery Environment: Gateway to Discovery, ACM.

Sattrup, P.A., 2012. Sustainability - Energy Optimization - Daylight and Solar Gains. Det Kongelige Danske Kunstakademis Skoler for Arkitektur, Design og Konservering.

SBI, 2013. Be10.

See, R., Karlshoej, J., Davis, D., 2012. An Integrated Process for Delivering IFC Based Data Exchange 53.

Seppanen, O.A., Fisk, W.J., 2006. Some Quantitative Relations between Indoor Environmental Quality and Work Performance or Health. ASHRAE Res. J. 37–41.

Sheikh, M. El, Gerber, D.J., 2011. Building Skin Intelligence A parametric and algorithmic tool for daylightning perfromance design integration. Proc. 31st Annu. Conf. Assoc. Comput. aided Des. arcitecture.

Shi, X., 2011. Design optimization of insulation usage and space conditioning load using energy simulation and genetic algorithm. Energy 36, 1659–1667.

Shi, X., Yang, W., 2013. Performance-driven architectural design and optimization technique from a perspective of architects. Autom. Constr. 32, 125–135.

Simon Flöry, Schmiedhofer, H., Reis, M., 2015. Goat.

Solar Heating & Cooling Programme, 2010. IEA Task 41 State-of-the-art of digital tools used by architects for solar design, Solar Energy and Architecture.

Solema, 2013. DIVA / Viper [WWW Document]. http://www.food4rhino.com/project/diva-rhino-0.

Souza, C., Knight, I., 2007. Thermal performance simulation from an architectural design viewpoint. Build. Simul. 87–94.

Stiny, G., 1980. Introduction to shape and shape grammars. Environ. Plan. B, Plan. Des. 7, 343–351.

Struck, C., 2012. Uncertainty propagation and sensitivity analysis techniques in building performance simulation to support conceptual building and system design. Eindhoven University of Technology.

Struck, C., de Wilde, P.J.C.J., Hopfe, C.J., Hensen, J.L.M., 2009. An investigation of the option space in conceptual building design for advanced building simulation. Adv. Eng. Informatics 23, 386–395.

Sullivan, R., Arasteh, D., Papamichael, K., Kim, J. j., Johnson, R., Selkowitz, S., McCluney, R., 1988. An indices approach for evaluating the performance of fenestration systems in nonresidential buildings. ASHRAE Trans. 94.

Sutherland, G., Maldonado, E., Wouters, P., Papaglastra, M., 2013. Implementing the Energy Performance of Buildings Directive (EPBD).

Saaty, T.L., 1977. A scaling method for priorities in hierarchical structures. J. Math. Psycology 15.

Saaty, T.L., 2008. Relative measurement and its generalization in decision making why pairwise comparisons are central in mathematics for the measurement of intangible factors the analytic hierarchy/network process. Rev. la Real Acad. Ciencias Exactas, Fis. y Nat. Ser. A. Mat. 102, 251–318.

Saaty, T.L., Zoffer, H.J., 2013. Principles for Implementing a Potential Solution to the Middle East Conflict. Not. AMS 60.

Thermal Energy System Specialists (TRNSYS), L., 2013. Transient System Simulation Tool.

Thuesen, N., Kirkegaard, P.H., Jensen, R.L., 2010. Evaluation of BIM and Ecotect for conceptual architectural design analysis. Proc. Int. Conf. Comput. Civ. Build. Eng.

Tonel, G., 2007. Unconstrained optimization using Hooke & Jeeves, MATLAB [WWW Document]. http://se.mathworks.com/matlabcentral/fileexchange/15070-unconstrained-optimization-using-hooke---jeeves?s_tid=srchtitle

Torczon, V., 1997. On the Convergence of Pattern Search Algorithms. SIAM J. Optim.

Toth, B., Janssen, P., Stouffs, R., Chaszar, A., Boeykens, S., 2012. Custom Digital Workflows: A New Framework for Design Analysis Integration. Int. J. Archit. Comput. 10, 481–500.

Toth, B., Salim, F., Burry, J., Frazer, J., Drogemuller, R., Burry, M., 2011. Energy-Oriented Design Tools for Collaboration in the Cloud. Int. J. Archit. Comput. 09, 339–360.

Trimble, 2013. SketchUp.

Triple E, Ecofys, TNO, 2014. ENER/C3/2012-436 – Market study for a voluntary common European Union certification scheme for the energy performance of non-residential buildings. Rotterdam.

Trubiano, F., Roudsari, M.S., Ozkan, A., 2013. Building simulation and evoulutionary optimization in the conceptual design of a High-Performance office building. In: 13th Conference of International Building Performance Simulation Association. Chambéry, France, pp. 1306–1314.

Turing, A.M., 1936. On Computable numbers, with an application to the Entscheidungsproblem.

Turrin, M., von Buelow, P., Kilian, A., Stouffs, R., 2012. Performative skins for passive climatic comfort. Autom. Constr. 22, 36–50.

U.S. Department of Energy, 2013. Energy+.

U.S. Department of Energy, 2015. Building Energy Software Tools Directory [WWW Document]. URL http://apps1.eere.energy.gov/buildings/tools_directory/ (accessed 1.1.15).

Ulrich, R.S., Zimring, C., Zhu, X., DuBose, J., Seo, H.-B., Choi, Y.-S., Quan, X., Joseph, A., 2008. A review of the research literature on evidence-based healthcare design. Heal. Environ. Res. Des. 1(3).

United States Environment Protection Agency, 1986. Report to Congress on indoor air quality: Volume 2.

UTO, 2013. Geco [WWW Document]. http://www.food4rhino.com/project/geco.

Uzoka, F.-M.E., Obot, O., Barker, K., Osuji, J., 2011. An experimental comparison of fuzzy logic and analytic hierarchy process for medical decision support systems. Comput. Methods Programs Biomed. 103.

Várkonyi, V., 2010. Next Evolution of BIM [WWW Document]. URL http://www.aecbytes.com/viewpoint/2010/issue_53.html (accessed 3.4.13).

Viehland, D., Hardy, J., Deville, J., Schementi, J., Rome, K., Icaza, M. de, Earl, A., Matousek, T., 2015. IronPython the Python programming language for the . NET Framework [WWW Document]. URL http://ironpython.net/ (accessed 4.30.15).

Vierlinger, R., 2014. Octopus. http://www.grasshopper3d.com/group/octopus.

Visscher, H., Meijer, F., 2006. Housing in an expanding Europe: theory, policy, participation and implementation. In: Building Regulation for Housing Quality in Europe. Ljubljana, Slovenia, pp. 1–15.

Wang, W., Zmeureanu, R., 2005. Two-Phase Application of Multi-Objective Genetic Algorithms in Green Building Design. Ninth Int. IBPAS Conf. 1323–1330.

Wang, W., Zmeureanu, R., Rivard, H., 2005. Applying multi-objective genetic algorithms in green building design optimization. Build. Environ. 40, 1512–1525.

Ward, G.J., Rubinstein, F.M., Clear, R.D., 1988. A ray tracing solution for diffuse interreflection.

ACM SIGGRAPH Comput. Graph. 22, 85–92.

Wetter, M., 2011a. Generic Optimization Program User Manual. Energy 1998–2011.

Wetter, M., 2011b. A view on future building system modeling and simulation. Build. Perform. Simul. Des. Oper. 1–28.

Wetter, M., Polak, E., 2004. A convergent optimization method using pattern search algorithms with adaptive precision simulation. Build. Serv. Eng. Res. Technol. 25, 327.

Wilde, P. De, Voorden, M. Van Der, Brouwer, J., Augenbroe, G., Kaan, H., 2001. The need for computational support in energy-efficient design projects in the netherlands . Delft University of Technology , Faculty of Architecture , Building Physics Group Delft , 2600GA - The Netherlands 2 . Georgia Institute of Technology , College 513–520.

Wilkinson, S., 2011. Stochastic Algorithms for Insolation Optimisation. Comput. Synth. BENVGACC.

World Health Organization, 2006. WHO Air quality guidelines for particulate matter, ozone, nitrogen dioxide and sulfur dioxide: global update 2005: summary of risk assessment. Geneva World Heal. Organ. 1–22.

World Health Organization, 2012. Burden of disease from Household Air Pollution for 2012 35, 17.

Wurzer, G., 2010. Schematic Systems-Constraining Functions Through Processes (and Vice Versa). Int. J. Archit. Comput. 8, 201–217.

Yezioro, A., Dong, B., Leite, F., 2008. An applied artificial intelligence approach towards assessing building performance simulation tools. Energy Build. 40, 612–620.

Zarzycki, A., 2010. Exploring parametric BIM as a conceptual tool for design and building technology teaching. In: Proceedings of the 2010 Spring Simulation Multiconference on - SpringSim ’10. ACM Press, New York, New York, USA, p. 1.

Zhai, Z., 2003. Developing an integrated building design tool by coupling building energy simulation and computational fluid dynamics programs. Massachusetts Institute of Technology.

Zhang, J., Michael Pelken, P., Chen, Y., Rice, D.J., Meng, Z., Semahegn, S., Gu, L., Henderson, H., Feng, W., Ling, F., 2013. “Virtual Design Studio”—Part 2: Introduction to overall and software framework. Build. Simul. 6, 253–268.

Zitzler, E., Laumanns, M., Thiele, L., 2001. SPEA2: Improving the strength Pareto evolutionary algorithm, Swiss Federal Institute of Technology (ETH) Zurich, TIK-Report 103.

List of figures

List of Tables

Appendices

Journal Articles

Appendix A.1

Negendahl, K., 2015. Building Performance Simulation in the early design stage: An introduction to Integrated Dynamic Models. *Automation in Construction*, 54, pp.39–53.

Appendix A.2

Negendahl, K. & Nielsen, T.R., 2015. Building energy optimization in the early design stages: a simplified method. *Energy and Buildings,* 105, pp. 88-99 .

Conference articles

Appendix A.3

Negendahl, K., 2014. Parametric design and analysis framework with integrated dynamic models. In *Proceedings of the 3rd International Workshop on Design in Civil and Environmental Engineering*.

Appendix A.4

Negendahl, K., Perkov, T. & Heller, A., 2014. Approaching Sentient Building Performance Simulation Systems. In *Proceedings of eCAADe 2014*. Newcastle, UK.

Appendix A.5

Negendahl, K., Perkov, T. & Kolarik, J., 2015. Agent-based decision control - how to appreciate multivariate optimisation in architecture. In *5th Design Modelling Symposium - Modelling Behaviour*. Copenhagen, Denmark.

Confernece Poster

Appendix A.6

Negendahl, K., 2014. Parametric City Scale Energy Modeling Perspectives on using Termite in city scaled models. In *iiESI European Workshop*. Kgs. Lyngby, Denmark.

Survey example

Appendix B.1

Survey results

Appendix B.2

Automation in Construction 54 (2015) 39–53

Contents lists available at ScienceDirect

Automation in Construction

journal homepage: www.elsevier.com/locate/autcon

Review

Building performance simulation in the early design stage: An introduction to integrated dynamic models

Kristoffer Negendahl *

Technical University of Denmark, Department of Structural and Civil Engineering, DTU Byg Brovej, Bygning 118 2800 Kgs. Lyngby, Denmark

A R T I C L E I N F O

Article history:
Received 6 November 2013
Received in revised form 21 February 2015
Accepted 4 March 2015
Available online xxxx

Keywords:
Integrated dynamic model
Parametric
Dynamic
BIM
BPS
Building performance
Building energy
Collaboration
Early design stages

A B S T R A C T

Designing with building performance simulation feedback in the early design stage has existed since the early days of computational modeling. However, as a consequence of a fragmented building industry building performance simulations (BPSs) in the early design stage are closely related to who is creating and operating the BPS models. This paper critically reviews the different ways designers and analysts use BPS in the early design stage. One of the key findings is that most tools and methods used in the early design stages are insufficient to provide valid feedback while in the same time being flexible enough to accommodate a rapid changing design process. The main concern points to the way geometrical models and analytical models are combined and how this affects the way the buildings are designed and perform. This paper concludes that integrated dynamic models may combine a design tool, a visual programming language and a BPS to provide better support for the designer during the early stages of design as opposed to alternatives such as the current implementation of IFC or gbXML or the unaccompanied use of simulation packages.

Contents

* Tel.: +45 45251700.
E-mail addresses: kristoffer.negendahl@gmail.com, krnj@byg.dtu.dk.

http://dx.doi.org/10.1016/j.autcon.2015.03.002
0926-5805/

1. Introduction

Designing energy efficient buildings with good indoor environment involves elements of expertise deriving from multiple disciplines such as architects, civil, mechanical and electrical engineers. With current emphasis on sustainability, including building energy and indoor environment, design requirements from the involved disciplines have become more important in the early design stages.

As a consequence building performance simulations (BPSs) are increasingly used to design buildings.

While numerous unified tools that act both as a design tool and BPS tool exist, building designers still seem to prefer to create and explore design options in dedicated design tools such as ArchiCad, Sketchup, Revit, Rhino, and Maya, as they support the concept of a sketch and the freedoms associated with design tools [1]. As a result the most prevalent method of receiving performance feedback in the early design stages is associated with either manually (re)modeling the designs in dedicated BPS tools or with a manual import and export task of the geometry. The import/export process works with proprietary formats or common data schemes such as IFC (Industry Foundation Classes [2]).

During the last few years new ways of integrating design tools and BPS tools at runtime-level have been developed. These new methods provide performance feedback directly in the native design tool and opens up for new design scenarios previously inaccessible for architects and engineers during early design stages.

The integration of a design tool and a BPS tools is fundamentally changing building design into a faster, performance-aware and more flexible process, which eases the production of multiple design alternatives. The posing question is, how do these coupled models fit into the design process of the early design stages?

To answer this one must first deduct the ways design tools and BPS tools can be coupled, however before one can assess the coupling choices, it is necessary to define the users and their requirements for these tools. Acknowledging that requirements of building design are comprised of quantitative elements (i.e. yearly consumed energy, amount of daylight, cost etc.) and qualitative elements (i.e. social impact, spatial planning, esthetics, etc.), building design aims to satisfy multiple criteria beside measurable performances. This implies that building design is evidently connected to role-definitions and collaborative processes, and it also implies that the utilization of building performance has to respect the broad extent of both quantitative and qualitative elements of building design. This article reviews current interdisciplinary collaboration in the early design stages and specifically reviews, how building performance simulations are used by architects and engineers in designing buildings. The second part of the review addresses the main developments in which a design tool and a BPS tool are combined. Under these circumstances using BPS environments in the early stages of design propagates in two main questions:

1) Who operates geometric models and building calculation models?
2) What is the best way to couple geometric models and building calculation models in the early design stages?

These questions are explored in two dedicated parts in the article.

2. Model operation — the users, the geometry and building performance

Operating a geometric model[1] implies both the creational process of making geometry and the direct effect on building performance. Operation of a geometric model infers a) creating unambiguous geometry that represents a building, and b) any changes, modifications and manipulation during the design process is performed by the model operator. The typical user or model operator consists of an architect creating and manipulating a geometric model in a design tool. The term design tool covers any tool that is able to represent building geometry. Another typical model operator is the engineer, who in similar ways creates and controls a calculation model in a BPS tool. The amount and quality of human and machine interaction between different model operators form convergence between the operated models.

Separated but correlated models have been the standard procedure, when architects and engineers have designed and later constructed buildings. Separated geometric and calculation models is one of the many symptoms of a disciplinary fragmented building industry. Souza [3] prefers to describe the disciplinary division by the qualification of a person, rather than the background of the person, thus classifying two main roles in building design as the building designer and the simulationist. It may be appropriate to let the building designer operate design tool and simulationist operate the BPS tool, but in many cases the roles and the operations of the models are less clearly defined.

The introduction of building information modeling, BIM (specifically referring to the gbXML [4] and IFC [2] standards), sought improvement, when teams are working with separated models. Even though the concept of a common reference model makes sense in all stages of building design, the early stages are often detached from any form of building information model. Seen from a technical point of view this is mainly due to the fact that many of the tools (both design tools and BPS tools are yet to implement resilient tool integration).

This article examines, to what extent building performance simulation software is integrated in the early design stage, thus touching upon two domains of integration: user integration (Fig. 1.1) — concerning human collaborative interactions, and model integration (Fig. 1.3) — concerning higher levels of computational automations between design tools and BPS environments. The third domain; tool integration (Fig. 1.2), mainly concerns technical details of specific tool interoperability and is omitted in this article.

2.1. The users — model operation

Model operation is today primarily a collaborative concern, which refers to 'whom' and 'what' is to be manipulated, rather than 'how' the model is manipulated.

Different collaborative partnerships have been suggested, examined and documented over the past years. Researchers such as Attia [5], Banke [6] and Hermund [1] have documented the current relationship and effect of using design tools linked to BPS through interviews and surveys among architects and engineers. Some of their results on

[1] Geometric models refer explicit to building geometry in computational geometry and topology in geometric modeling and graphics.

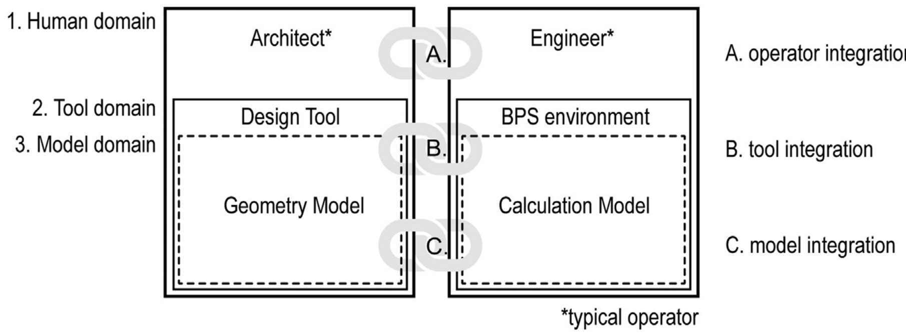


Fig. 1. Three types of integration: 1) integration in the human domain, 2) integration in the tool domain, 3) integration in model domain.

operator integration and tool integration are discussed throughout the review. Other researchers e.g. Klitgaard et al. [7] and Petersen [8] have described either ideal or future collaborative relationships in the early design stages when seeking to encounter and improve building performance. Based on Klitgarrd et al.'s and Petersen's research the three most prevailing collaborative arrangements are discussed here:

1) The engineer as an assistant to the architect.
2) The hybrid practitioner acting both as engineer and architect.
3) The engineer as a partner to the architect.

2.2. The engineer as an assistant to the architect

The traditional role distribution during the early design stages is directing a horizontal linear process, where knowledge is transferred from the architect to the engineer. In such arrangements or approaches the engineer is acting assistant (often in a narrow field of expertise) for the architect.

Klitgaard et al. [7] describe and discuss the problems of the traditional collaborative approach in the early design stages. The engineer and the architect respectively control their own domain and consequently their own toolsets. This implies that all building performance evaluation is done by the engineer, who later consults the architect on the matter of eventual design modifications. Such arrangement generates high risks of model divergence and can lead to all sorts of errors and misunderstandings during the early design stages. All communication is done in the human interaction domain and model interaction is reserved to exporting geometric data to the BPS environment. As the models are completely separated, no direct feedback options are available.

To counter some of the misconceptions and problems associated to the human interaction domain, Mora et al. [9] describe a guideline for an assisting engineer. Their study suggests different principles of how assistance is to be handled. Responsiveness from the assistant and the ability to illustrate alternatives is important. Assisting also means not to interfere in the creational processes of the design, but rather assist in analyzing outcomes of the design.

Other researchers (e.g. Struck et al. [10]) argue that the assisting engineer may pose a more active position in the operator domain. In this case the role of the engineer is broadened to support the exploration of the decision space for selection of most suitable design alternatives. This means that the good assisting engineer would think laterally within the decision space and in alternative ways of solving building design, thus thinking beyond a particular solution suggested by the architect. When the engineer generates design alternatives, the challenge is to assist design, rather than automate design, that may allow spontaneous, creative and flexible processes that acknowledges the expertise of the participating design disciplines [10].

As the engineer seeks out alternative solutions, she is tied to the calculation model in the BPS environment. Geometry, system setups and other influential parameters of the building design can be modified to solve or improve certain performance-related objectives. The risk of modifying the building design in a way that is uncorrelated to the architectural concept and intent becomes more profound, as the engineer isolates the building design within the calculation model (Fig. 3).

The reconstructed calculation model is rarely represented in the same details, as it was in the design tool, mainly because of the BPS tool limitations in handling advanced geometry. In some cases the calculation model is simplified on purpose, because detailed modeling adds to the simulation time but does not affect the results significantly. The creational position of the engineer as calculation model operator is important when trying to get meaningful results from the model within a limited time frame. But the act of interpreting the geometrical model may sidetrack the architect in the process, since the original model in the design tool is no longer the basis of investigation. To recalibrate two models the engineer must interpret the calculation model results and communicate, how the original model performs, which in many cases leads to suggestions of, how architects should modify their geometric model. Subsequently the user integrations within the human interaction domain determine, how much information is transferred from the BPS environment to the design tool. Various problems are associated with this approach. Klitgaard et al. [7] mention cost increase in terms of man hours related to design iterations and numerous modifications that may not be approved by either participant.

2.3. The engineer as an architect, the architect as an engineer, the hybrid practitioner

The way buildings are designed changes, as emerging digital technologies become a part of the architectural design process. Some architects even talk about a development which will radically change our perception of architecture [11]. Holst argues that it is the responsibility of the designer to have an overview and control of the process in terms of the esthetic, functional and technical requirements and the design intentions and needs. Areas which today are divided between architects and engineers must in the hybrid approach be practiced by only one stakeholder. To ensure a process based upon augmented choices rather than serendipity, the hybrid practitioner must possess the ability to combine different ways of thinking, bridging the intuitive and the formal. The practitioner thereby ensures satisfaction from an architectural as well as an engineering perspective. This type of designer has equal insight in technical analyses as well as architectural design.

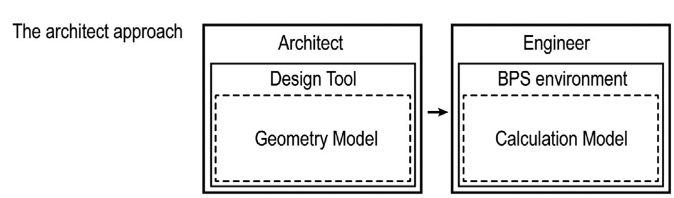


Fig. 2. The architect approach as based on Klitgaard et al. [7]. The engineer acts as assistant to the architect.

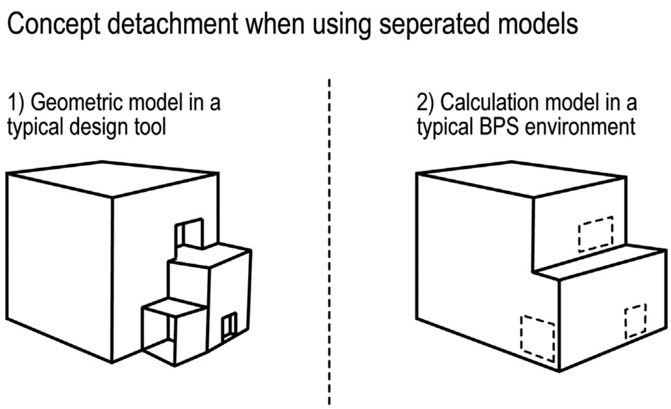


Fig. 3. Manual reproductions of models between design tools and BPS environments result in poor model convergence.

Several inspiring attempts to bring architecture and engineering together in the early stages of design have been done over the past few years. Through formulation and implementation of multi-objective optimization methods Kataras [12] has explored the building geometry with respect to the incident solar irradiation. Kataras is one among increasingly many architects actively using constraints as design-drivers for human design exploration. Other researchers such as Welle et al. [13] do not separate the roles of the architects and engineers in the design process, in which technical and analytical background seems irrelevant. Taking this view on the process makes it far more plausible to remain problem focused rather than the usual approach of being solution focused. The hybrid practitioner embraces rational optimization methods and other algorithmic automations (discussed in Section 4). Shi [14] argues that multivariate optimization, which is a typical engineering discipline, is a skill not too advanced and can be learned. That architects with the right tools in hand may be used to solve more general architectural design problems and building performance is just one of many multiple objectives that need careful attention.

Killian [15] discusses the level of expertise required for the hybrid practitioners to be able to sufficiently do their job. One example he mentions in practice is Foster and Partners, whose expertise is turned into computational procedures. The embedded expertise ensures, for instance, that certain geometrical requirements needed for fabrication are met [15]. It is likely that an adequate amount of embedded expertise, when provided by a combination of a geometric model and results from a calculation model, will deliver sufficient knowledge to make up for the lack of knowledge held by the practitioner. However, embedded expertise, when provided from a computer simulation, does not necessarily ensure sufficient analytical experience of the practitioner.

2.4. The engineer as a partner to the architect

Integrated design processes such as (IDP), specifically referring to IEA's Task 23 [16], suggest a collaborative approach of operating the early design stages — that is seeking to outline, who is to operate, and what is to be operated in a model. The method defines a design team composed of actors, where actors refers to all project participants who have a relevant influence on the content and course of project design and realization [16]. This determines the necessity of an integrated

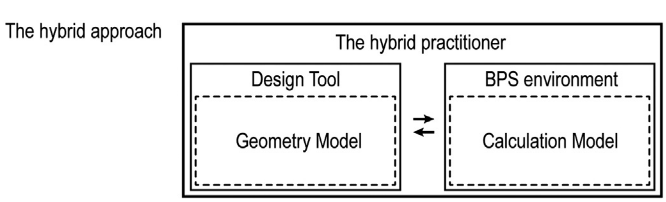


Fig. 4. Example of the hybrid practitioner approach.

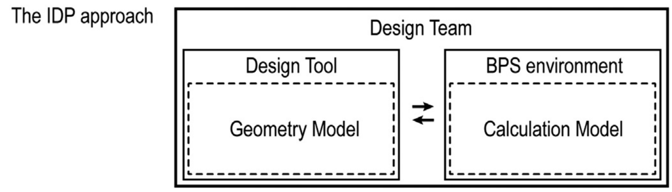


Fig. 5. Example of an Integrated Design Process (IDP) approach.

model operated by several contributors, including architects and engineers. While the two first collaborative approaches do not aim to define a common goal, the IDP process has defined frameworks to negotiate criteria and end goals. Some argue that Integrated Project Delivery (IPD) is equivalent to IDP with little variances such as the fact that IPD associates project procurement and a predefined partnership among collaborators with the appropriation of design data through BIM [17]. Therefore this article will not differ between the two approaches. One of the reasons, why early decision making should include design team expertise, is that cost of change during the design stages is rapidly increasing. Researchers such as Löhnert et al. [16] have identified this circumstance and points out the large potential of savings when attaining more knowledge of building performance in the early design stages.

IDP defines that the architect, the engineer, client and other participants are provided control of building design as expressed in Fig. 5. IDP is defining many more lateral connections across disciplines as opposed to the traditional hierarchical ones, allowing complex design problems to be addressed collaboratively and efficiently. Frameworks for implementing process driven integrations like IDP such as MACDADI [18] and VDS [19] have been suggested. Arguably this leads to better performing buildings, however general implementation in practice of such frameworks has yet to be documented.

The integrated design process faces many complications in terms of collaboration on model level. Nevertheless, far from all collaborative concerns can be solved in technical terms. Mainly differences of model operator intentions and objectives cannot be solved at model level.

Supporting different model operators will however ease information and data sharing within the design team. The typical architect and the typical engineer possess (at least slightly) different approaches in operating the model. An integrated model[2] has to accommodate a spectrum of targets and end objectives fitting all the contributors in the design team. As this article investigates more methods utilizing IDPs, it is worth mentioning that the practical relationship between architects and engineers today may be closer to the traditional architect approach than to the IDP or the hybrid approach.

2.5. Conflicts in collaborative modeling

When different disciplines work together conflicts arise. A conflict between architects and engineers has been discussed since the separation of the two disciplines. Many other researchers, e.g. Banke [6] and Hermund [1], have explored these problems in terms of social and sociotechnical circumstances, many of the problems are identified and related to the human domain as depicted in (Fig. 1.1). It is important to mention some of the human collaborative challenges to explain the computational collaborative challenges described in Section 3 of the article. Byrne [20] explains the root of conflict as differences of motivations between the architect and the engineer:

> Architects evaluate all aspects of the design, from broader issues of internal and external relationships to more detailed aesthetic measures such as material use, texture and light. Engineers evaluate the integrity of the structure itself. To oversimplify, architects are concerned with spaces, engineers are concerned with forces.
>
> [[20]]

[2] Integrated model: a model that is coupled across a design tool and a BPS environment.

As Byrne oversimplifies both the engineers' and architects role, he notes the real conflict related to the model domain is concerning form and function:

> Once the form has been chosen, the design process focuses on satisfying the constraints of the original design specification. At the centre of this process there is a conflict between form and function. While these two attributes of a design are not mutually exclusive, there can be a trade off when realizing a design.
>
> [[20]]

2.5.1. Tradeoffs

Tradeoffs are pivot for many researchers in the field. However tradeoffs can only be accurately estimated, if all consequences of trading one solution for another are known. In collaborative situations this means that tradeoffs must be known to all members in the team. In other words: every attributes of design must be visible and known to all parties in the design team, before a form and function is definite. This may be a very difficult task to solve. The challenge derives from the design teams' disposition of end objectives as well as understanding the effect of tradeoffs. Having mutually different objectives may be the main source of potential conflicts. One challenge in particular, mentioned by Byrne [20], concerns the form and function as being inseparable from building energy performance and indoor environment, while in real life, objectives of some members in the design team are biased towards either form and function or energy performance. Although performance covers everything from energy performance to social performance, the balancing act of form and function is equivalent to a balancing act of quality defined and quantity defined performance. Performances are undeniably interrelated.

2.5.2. Objectives and value drivers

Value drivers that are characterized as discipline or project specific scaling of a building project success [21], which in a sense is closely coupled not only with client expectations, but also with more-lower level agendas, such as code compliance. The preference of value drivers is very important when seeking solutions of good building performance with a low cost, and it is presumed that affection to certain objectives appears, when architects and engineers prioritize the importance of value drivers. This has been proved by Attia et al. [22] when looking into the engineers' and architects' preference of, what BPS software can deliver to the user. Architects in general are concerned with architectural design issues including shading, passive heating, orientation, natural ventilation and geometry. HVAC systems and controls were ranked by architects as the least important, whereas engineers in general ranked HVAC system, controls, comfort, glazing openings and insulation respectively in the top [22]. It may be argued that these issues are relevant to different stages in the design process, i.e. the orientation is to be decided in the conceptual stage and the HVAC system settings in the detailed design stage, in which case the difference in objectives may not be a problem in itself. Yet every issue is part of a coupled problem, and it cannot be solved alone. If the value driver is about building performance, both HVAC systems and orientations may have to be considered simultaneously. The matter of when different objectives should be handled is discussed in Section 4.3.

Consultants have sought to improve their influence on the early design by defining a framework for design, typically by limiting the design space of the practicing architect. The question of, who is the main model operator, is still an active open debate. On one side is the classic architect arguing that the holistic expression must remain with the architect in order to protect the architectural value of the building, e.g. manifested in Henning Larsen Architects Integrated Energy Design, IED-approach:

> IED is a holistic method. Technological developments have resulted in an increased specialization and fragmentation of knowledge making it difficult to view the project and its connecting functions as a whole. However, high-technology knowledge is not unwanted in integrated design, which seeks to understand the function of the entire system instead of just looking at the technical answer.
>
> [[23]]

On the other side the classic engineer argues that building energy codes and demands on indoor climate demand a stronger position in the earliest phase of the design, since site, geometry and orientation are crucial factors of the building performance. Alectia describes the IDP-approach:

> The method (Integrated Design Process) relies on the use of building simulations to illustrate how design parameters affect energy consumption and quality of the indoor environment before actual design decisions are made.
>
> [[24]]

The main difference between the two approaches refers to, who the operator of the design is, and what the central end goal of the particular design is.

Architects tend to look further and broader than the practicing engineer, hence seeing the project in a holistic sense. Engineers on the other hand are often better suited in the terms of understanding individual building functions and controlling these in relation to quantified building performances.

Further reading in the differences between architect and engineers' design thinking can be found in a recent review by Souza [3].

2.5.3. Legislations, cost and client requirements

Even if the main architectural goal is not derived directly from building performance (energy and indoor environment), the buildings still need to conform to minimum standard classifications and government imposed legislations, often known as requisite criteria. Typically the main objective of engineering is to follow requisites at minimum cost. When conflicts of requirements involve legislations, the conflict tends to concern minimizing the construction expenditures of fulfilling the legislations rather than minimizing energy consumption and improving indoor environment at any cost (including life time expenditures). Client requirements are equally important and may often function as ultimate objective requisites, however many clients do not know the actual consequence of their requirements. Soebarto [25] mentions the client's preference e.g. being close to the outdoor environment is more important, and therefore having large openings is more important than having a thermally comfortable space all the time. It is of this reason the building designers' (or design teams') responsibility to weigh and interpret client requirements and inform clients of the consequence of their requirements. Only an informed basis enables the client to understand, what they might be asking for.

2.5.4. Performance-based design

It is to be noted that some architects practice architecture that derives directly from building performance evaluations, often called performance-driven architecture or performance-based design defined by Kalay [26]. Shi and Yang [27] argue that:

> Compared with the conventional architectural design methodology, which focuses on space and form, performance-driven design takes a holistic view towards ecological and environmental performances of buildings while ensuring that the functions and esthetics of the design are not overlooked.
>
> [[27]]

The concept of form that follows performance is implemented in various degrees, ranging from performance information that inspires geometry to more excessive approaches, where form and function is generated directly from evolutionary performance evaluations like

Pasold and Foged [28]. Another similar design ideology that incorporates building performance is the evidence-based design approach. It is noted that Kalays [26] definition of performance is not explicitly defined as building energy, cost and climate, but a measure of the desirability of the predicted behavior of a design solution. For that reason performance can cover non-conventional performance, such as fulfilling social sustainable criteria — or any quality defined and quantity defined performances.

2.6. Summary on users and model operation

Operating models are shifting from being a computer- and programmer-centric discipline to a more interactive and more intuitive process, thus making building performance evaluations and calculations more attainable for non-simulation-specialists. Nevertheless it is possible to divide the model operators into two categories: 1) those that operate the geometry, and 2) those that operate the building performance simulations. Three variants of operational approaches are identified. The architect approach, the hybrid approach and the IDP approach. Considering the majority of simulation applications are targeted engineers and have limited capacity to interface design knowledge across disciplines [29], the BPS are being used in new ways seeking to inform better building design. Driven mainly by architects, as described by Banke [6], building performance has come within reach of the architect and is now both informing and affecting early design decisions. The role separation between engineers and architects are blurring, yet differences between architects and engineers affect how models are constructed and operated in design tools and BPS tools.

A recent survey performed by Attia et al. [22] documents the typical differences between architects and engineers as main operators of a calculation model. The survey showed that architects value fast analysis feedback and support for decision making as well as giving the architect confidence in creating real sustainable design. Today most BPS environments do not support the architect as described. BPS environments are built by engineers to accommodate engineers, who supposedly all have a fundamental understanding of the technical areas in play. Klitgaard et al. [7] argue that most BPS environments are developed to accommodate verification of building design rather than exploring building design. This means that most BPS tools are developed for later design stages, such as code compliance in detailed design, and therefore difficult to use in early design decisioning processes. These are some of the main challenges, when users are operating BPS environments in ways not initially intended.

Model operation on the basis of informed BPS results needs careful handling and knowledge in building physics. One of the biggest challenges is to maintain credibility of the building performance evaluations regardless of, who performs the evaluation. The inclusion of a specialist in the early design stages is paramount to achieve a valid ground of informed design, or as Ochoa et al. [30] put it: Input quality affects accuracy while output needs careful expert interpretation. As a consequence of those changes, engineering as a discipline is approaching the early design stages.

3. Model integration — geometric models and calculation models

3.1. The tools

Recent papers [27,31–33] examine, how a geometric model dynamically can be operated in relation to building performance simulations. Processing a geometric model from design software to a BPS environment has often been associated with a manual export/import task. Some of the main concepts are to ease the operation of building performance simulations by controlling the geometry and the simulation model from a native design tool[3] (as seen from an architects' point of view). One idea is to provide the building designer with results from simulations from an external BPS source inside the design tool to better inform the designer in the process of designing the building. Other concepts consist of operating the geometrical model in the very same environment that facilitates the building performance simulations, thus merging the geometric model and the calculation model via a unified design and calculation tool. Yet some researchers seek to unify the design tool and the BPS tool by defining a common exchange format.

What is of interest in this section of the review is, how design tools and BPS tools are coupled in the best way to support the design of better performing buildings. The definition of best support is established in the discussion (Section 4).

3.2. Ways of integrating design tools and BPS

There are three methods to integrate design tools and BPS in the early design stages (see Fig. 6).

3.3. Combined model method

Simulation packages, e.g. IESVE [34], contain a combined model (Fig. 7) and have the advantage of the operator being able to control the precision of the model within all steps of model production, manipulation and simulation. The combined model can handle both the modeling and the simulations at runtime level and provide consistency of the environment, which is an attractive feature for many users. The clear advantage of a combined model is that the design tool functionalities and the simulation tool functionalities essentially are integrated, thus enabling tool domain integration as depicted in Fig. 1B. The main disadvantage of this method is that the user is restricted to the options and features offered by a particular environment or program [35].

While combined models are limited to the functionalities of the modeling environment, most combined models support import and export geometry from other tools, but does not support dynamics of bi-directional updates between external tools. The concern of using simulation packages and combined models is that all involved participants must agree to use the same principal tool to maintain the high convergence between models. In this regard it may be difficult to use combined calculation models in larger uncoordinated groups and loose interdisciplinary projects.

3.4. Central model method

The central model (Fig. 8) is based on a widely used central framework concept most often referred to as BIM. The concept of centralizing building information data in a shared data schema is typically associated with the early influences of a buildingSmart initiative [36]. Various tools read and write to the same model and are thereby able to connect semantic information from a design tool to a BPS environment. Of recent methods based on IFC as a coupling medium, RenewBIM [37,38] and Simergy [39], and for gbXML as a coupling medium, OpenStudio [40], can be mentioned. Since design tools and BPS tools only recently have begun exchanging data through these types of open formats, there seem to be no consensus on, whether the design tool or the BPS tool should handle the convergence. Rose and Bazjanac [41] suggested to use an intermediate algorithm to create IFC-compliant space boundaries from a geometrical (CAD) model, thereby assisting the process of creating BPS friendly geometry. Other automated or semi-automated algorithms, such as automatic thermal zoning or simplification of sophisticated building geometry, will be needed to implement in either design tools, in BPS tools or in the IFC schema itself to make the coupling process between design tools and BPS tools automated.

The geometric model and the calculation model can in theory be dynamically coupled (bi-directional linked on runtime level) with an exchange file format, however this is rarely the case with the central modeling method used today. The main idea is unify the design tool

[3] Geometric models operated in dedicated CAD software is here classified as a design tool.

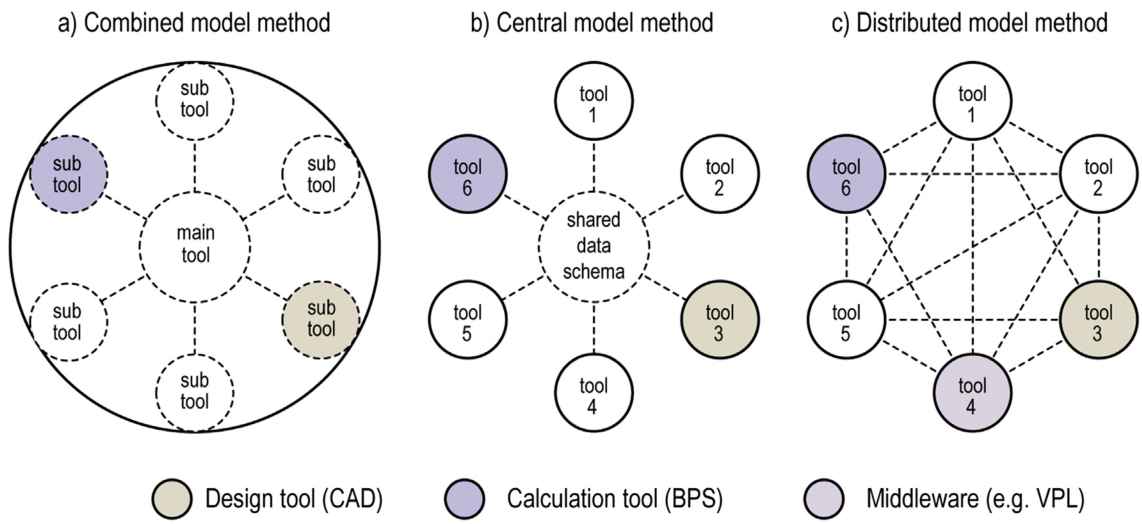


Fig. 6. Differences between coupling methods: a) combined model method (typically operated in a simulation package), b) central model method (using a central database/file format/schema), c) distributed model method (utilizing a middleware).

and the BPS tool by defining a common exchange format. Most frequently the tools using a central model are capable to exchange data with other dedicated software environments and are typically based on IFC or gbXML. Among these coupled or linked tools are BPS environments that are devoted to make energy and indoor environment performance calculations. Essentially the geometric model and the calculation model live in a shared format and every tool that supports the format is able to operate in the model. Implementations such as DCOM [42] are characterized by the reliance of a common data schema as the interoperability gateway between software, and therefore considered a central model method.

Moosberger [43] has shown examples of, how unidirectional IFC couplings to the BPS IDA ICE [44] can improve simulation convergence between the geometric model and the calculation model. This unidirectional method is widespread and does little to help provide the building designer with relevant performance feedback, unless the building designer is the operator of the calculation model. Plume and Mitchell [45] attempted to utilize the full centralized idea to perform different performance simulations. While IFC is capable of containing most of the data needed for the various BPS tools the building model needs to be constructed with collaborative interchange in mind and be able to anticipate the needs of design collaborators [45]. The central model has to be operated in consensus with all involved parties. As a consequence collaboration within centralized models has been considered time consuming and in some cases counterproductive in terms of design exploration. Using common data schema like IFC and gbXML to structure information exchange, regardless of whether it is proprietary or an open standard, imposes restrictions on, how designs can be described and thus explored [46]. Models translated from a shared building information model[4] are as precise, as the database or schema allows it to be. Limitations of the read and write structure derive from poor data quality of a single object, which will agitate through all of the connected environments of the database/schema. The main problem however, is presently not the open file formats but the lack of software support and user support of common open file formats.

Other methods that may be classified as a central model method is OpenStudio [40] which uses SketchUp as a design tool and couples the BPS Energy+ by its own file formats. Kalay suggested a method, which is often referred to as integrated collaborative design environments (ICDE's) [47]. The concept ranges beyond the interoperability with BPS through a central model when suggesting the inclusion of evaluation tools, negotiation tools and semantically rich databases. Elaborate semantic systems, as described by Kalay have been developed in different prototype forms over the past decade. An example is suggested by Wurzer [48], who successfully aided architects in automating process-planning, nevertheless no commercial product has yet been developed and used in practice.

3.5. Distributed model method

Van Nederveen et al. [49] argue that a distributed model fits better with the distributed responsibility and role definitions that are found in most collaborative practices.

> BIM environments are usually visualized as a circle of actors (disciplines, applications) positioned around a central Building Information Model. However, more and more people involved in BIM state that there is a need for a more decentralized approach. Not everyone believes in the ideal of a "central BIM" anymore.
>
> [[49]]

Distributed model methods (Fig. 9) can be seen as a response to the centralized model concept, disengaging itself from a top down control and one directional model operation. While it is important to note that architects' definitions of buildings in design tools do not necessarily reflect the needs of energy simulation [50], the tools may need to be able to adjust, conform, enhance and even eliminate parts of the model to be successfully interpreted by a BPS tool. This might be the largest problem of using a centralized model, as the model framework is depended on placeholder content that might never be created. Decentralizing the effort of creating content, however, will help model convergence. Nevertheless, decentralizing efforts of modeling may in worst case end up returning to the state of complete incoordination at model level (depicted in Fig. 4). Because of this, distributed models

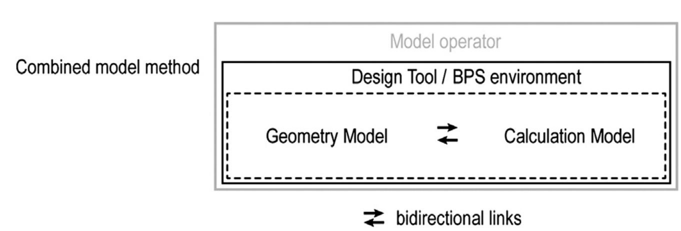


Fig. 7. Combined model. Consists of a design tool and BPS in the same environment.

[4] Referring to the two dominant implementations of BIM-standards on the market, gbXML and IFC.

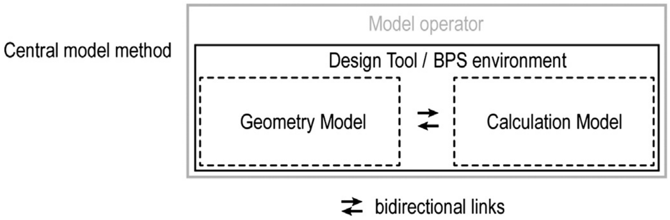


Fig. 8. Central model: a combination of a design tool and a BPS environment.

are characterized by deep integration at model level by utilizing a middleware component to translate data between the design tool and BPS tool.

Bazjanac [50] demonstrated that IFC is capable of providing sufficient interoperability in coupling Energy+ as a BPS, but a middleware element was necessary to exchange data. Additionally, he found that it was necessary to input secondary data required for the HVAC system, such as occupancy and use schedules. Since Bazjanac's investigations with IFC, various improvements and extensions have been added by the buildingSmart initiative. IFC2×4 now supports many of the then missing schedules and other technicalities missed in the interoperability investigations. Nonetheless, the improvements in the schemas do not change the fact that data is not generated by operators, and if data is generated, data may not be stored in the schemas in a way, which enables other tools to understand and use the data. The middleware is therefore not merely a simple converter between formats and platforms, but a system that is able to filter, modify and extend operator definitions to such a degree that the definitions reflect the needs of BPS environments (and obviously the needs of the design tools).

One of the recent developments in distributed model methods is Sustain [51], Virtual Design Studio (VDS) [19,52], and ZEBO [5]. Geometry models from design tools such as Revit [53], Rhino [54], and SketchUp [55] are imported through the middleware into the BPS. VDS [19], for example, functions as the necessary middleware to distribute and modify data to BPS tools, such as CHAMPS [56] and Energy+ [57]. VDS seeks to provide a hybrid tool made of components from architectural design practice and engineering simulation techniques and relates itself to existing assessment systems such as LEED, and DGNB, and at the same time incorporates advanced automation features, such as optimization algorithms and auto generation of e.g. VAV systems. While VDS supports various design tools, the actual model control and feedback are provided in the VDS system itself and as a result the all operators must have direct access to the system to react upon the feedback.

As the middleware is an essential part of a distributed model, the flexibility, features, and usability of the middleware is key to, how the model interoperability converge. An integrated dynamic model (Fig. 10) is a distributed model, where the middleware consists of a visual programming language (VPL).

An integrated dynamic model is basically a combined model composed of a geometric model controlled in a design tool dynamically coupled to a visual programming language (VPL), which is again dynamically coupled to a building performance simulation (BPS) environment. The middleware can be operated by either the simulationist [3], or the building designer, both of them or by a third, undefined operator.

VPL's such as Grasshopper [58], Dynamo [59], GenerativeComponents [60], Digital Project [61], and Yeti [62] are examples of some of the scripting tools, designers and architects are using to automate form generation. VPLs can in some cases be considered as design tools themselves, mainly because of the heavy use of geometric modeling functionalities. The reason why these tools are categorized differently than traditional CAD tools, is their ability to handle non-geometric data, and let operators create their own algorithms. The VPL is coupled bidirectionally with one or more design tools, e.g. Rhino [54], Revit [53] and MicroStation [63] and has direct runtime access to the design tool functions. VPLs coupled to design tools are able formalize to the exchange of data comprising of collections of geometric primitives, and the geometric-content-based data exchange of a VPL is in opposition to BIM's 'assigned-attribute-based' data structures [64]. This means that VPLs facilitate data across any content- and object relationship, while schemas like IFC prescribe object relationships through attribute data. The ability to cross-reference any relationships (both geometrical and non-geometrical relationships) provides a highly flexible and open environment. However the presence of VPL does not in itself guarantee interoperability between compatible software. The dependencies and rules of transferring data between tools must be defined in every integrated dynamic model.

Some of the recent frameworks that support the dynamic couplings between design tools, visual programming languages, and BPS are GenerativeComponents [60] combined with Design Link [65,66], which again delivers runtime couplings to Energy+ and Ecotect [67]. DEEPA [29] is one of the newer attempts that utilizes the VPL GenerativeComponents in combination with Energy+, in this case by using IFC to maintain import/export compatibility towards other coupling directions. Green Building Studio [68,69] is normally classified as a centralized calculation model, but since the framework is able to use the design tools Revit and Vasari [70] through the VPL Dynamo [59], together the tools can form an integrated dynamic model. The Rhino–Grasshopper-combination supports wide a range of couplings to various BPS. The facilitation of links to the BPS is handled by third party modules, such as Viper [71] (Energy+), ArchSim [72] (TRNSYS [73]), Geco [74] (Ecotect/Radiance), DIVA [71,75] (DAYSIM [76]) and Honeybee [77] (Energy+). All of these modules are coupled through the VPL Grasshopper to the design tool Rhino.

Davis [78] has compared the paradigms and scope of most common used VPLs in architecture listing some of the features and limitations of the different programming languages. Davis' main concerns are not tool-to-tool interoperability and performance related feedback, but focuses on user-to-tool support, as well as how the VPLs are built and applied in the architectural processes. As he describes, many architects have embraced the new tools, and examples of extensive utilization to generate form with VPLs are a trending phenomenon. Burry's [79] main argument of using VPL's in his book Scripting Cultures is the potential to free up the designer to spend more time on design thinking. Scripting and the use of VPLs are often associated with parametric automation of geometry in architecture, but geometry is just one aspect VPLs are able to automate. Some of the more advanced automations are found in the discussion. As long as the VPL is able to interpret the data, the VPL will not distinguish between data types. Toth et al. [46] see the elimination of the common data schema as a prerequisite for information exchange, allowing design freedom to create custom digital

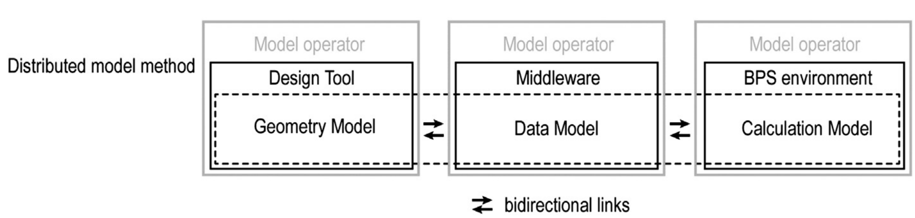


Fig. 9. Distributed model: a combination of a design tool a middleware tool and a BPS environment.

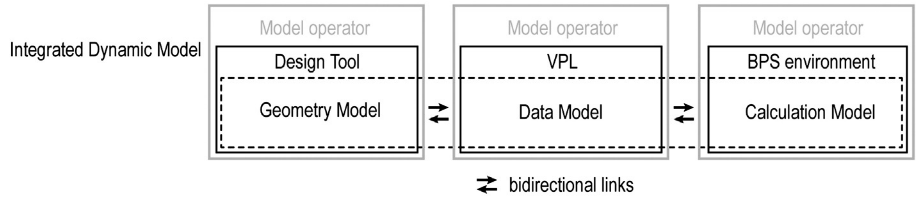


Fig. 10. Integrated dynamic model: a combination of a design tool, a VPL (visual programming language) and a BPS environment.

workflows unfettered by standardization constraints while distancing itself from the centralized model method. Toth et al. [46] further argue that the VPL does open up for exchanging data with IFC and other open formats, the VPL can so to speak act as a gateway to a common data schema.

3.6. Summary on integration of geometric and calculation models

In general design tools and BPS are independent and are difficult to integrate at runtime level. Three model methods have been characterized. The combined model essentially is both a design tool and a BPS tool, the central model method is a design tool coupled to a BPS tool over a standardized file format or schema such as IFC and the distributed model method facilitates a design tool coupled through a middleware to a BPS tool. Then, there is the variation of a distributed model, where the middleware consist of a VPL, which is called an integrated dynamic model.

The three model methods pose different benefits and limitations and may be preferred by different operators.

The best option seen from e.g. Hensen's [35] perspective seems to be: essentially two-way coupling of building energy simulation with separate software packages, where the programs provide the facility to link applications at run-time in order to co-operatively exchange information. In this way the linked environments are only limited to the mutual internal limits of sharing and translating data. Dynamic coupling of models has the main advantage of profiting on the precision of different BPS since updates can be bi-directionally implemented. As in the run-time coupling approach, a truly integrated simulation relies on the information exchange throughout the simulation to resolve a set of combined equations that represents the driving process of simultaneously occurring physical phenomena [80].

Essentially all three model methods support runtime couplings. Where the combined model has the highest level of convergence between the geometrical model and the calculation model, the central model has least convergence due to poor support of interpolation formats, and real runtime couplings is yet to be seen. The distributed model method, e.g. the framework of VDS [19], may support the same model convergence as suspected from a combined model. However distributed models such as VDS rely on one-way import from a design tool, which essentially shifts the design tool operation into the middleware. The distributed model method supports the most variations of BPS tools and delivers runtime couplings between the BPS and the design tool, while the variation integrated dynamic model seem to have best support of operator defined relationships between objects. Nonetheless integrated dynamic models require much individual scripting setting up object relationships when runtime linking a design tool to a BPS environment. This kind of work is automated without operator involvement in the combined model, while managed by the common formats such as IFC in the central model, or handled by the middleware in distributed models such as VDS.

4. Discussion

The central issue in this discussion is; what model method is better suited the early design stages, specifically when it comes to designing high performing buildings?

It can be rather difficult to argue which of the model methods is better. Each of the models can be evaluated in their ability to support features, precision and validity of creating trust worthy results from the coupled BPS. It may be speed of, or quality of relevant feedback provided by the operator. But the most important aspect might be how well the operator can benefit from the model environment, rather than being limited by it. In this case we need to go back to the question of, who the operator of the models is. Davis and Peters [64] argue that one of the most important features of the design environment (here model method) is its ability to provide design feedback that reflect the actual design and further argues; the worst case is when the designer is dissuaded from making the change and ends up with a design that was not so much created in their design environment as it was for the limitations of that environment [64]. As a result the model method must support wide range design tool functionalities, as well as various forms of BPS feedback and in addition provide valid and unambiguous results to the operator.

Based on Hopfe et al. [21], Attia et al. [22], Toth et al. [29] and Struck [81] the following criteria are identified in evaluating the model methods:

- Multidisciplinary model operation
- Model operator collaboration
- Model method flexibility
- Model method scalability
- Model method precision and validity
- Model method speed and visual feedback
- Model design support (features e.g. automation, optimization, and uncertainty analysis).

The model method evaluation concerns exclusively the geometry model and its specific attributes linked to one or more calculation models in one or more BPS environments. Additionally the operators of these coupled models must be seen in a generalized way to comply with both the concept of an exclusive design operator and the effort of a controlling design team as outlined in the previous part of the article. As a result the different operators can be represented by architects and/or engineers (or any other experts in the field of building design). The users may be collaborating (e.g. stated in the IDP) or even represented by the same individual (a hybrid practitioner).

In Attia's et al.'s surveys [22] of BPS and contradictions in engineers' and architects' needs, architects top-ranked BPS with abilities to be used in different design stages, as well as tools that incorporate a design tool (geometrical 3D modeler). Engineers specifically wanted improved interoperability through (and standardized use of) open formats such as IFC and gbXML, but did not ask for direct couplings with design tools. This difference in operator needs of BPS choices may also give rise to differences in operator needs in model methods.

4.1. Universal- or multi operator support

The first distinction between the three methods is the definition of model operators in relation to the model environment. The only model method that defines a clear operator segregation is the distributed model. The combined model as well as the central model method is constructed in such a way that only one operator (that obviously may consist of a team) is in control of the model. These model methods may be very appealing to small very integrated teams or solo-projects operated by hybrid practitioners. But larger teams without central and unified objectives (often associated with the architect approach) may see the unified operator-approaches as problematic and even counter-collaborative. On the other hand with segregated role definitions, the combined model and the central model may end up being operated by a few participants in the team, which again may provide better building performance, if the objectives are clearly performance related. The problem is that better building performance does not entitle a building to be a better building. Many qualities of design knowledge of the excluded part of the team may get lost in the process, when only part of a larger team is operating the model.

None of the three model methods fully supports the architect approach (Fig. 2), unless the architects themselves have direct access as model operators, or can state flawless human level integration with the BPS operators. Nevertheless, the distributed model is the only model that can separate role definitions on model level, which may leave the distributed model as the least-worst option of the model methods. All the models support the hybrid- and IDP operator approaches. Even though the models can support interdisciplinary design approaches, the model methods themselves do not ensure better collaboration. The mindset of the participants, the framework and features such as evaluating and negotiating criteria, solutions and results, do however improve collaboration. Nevertheless Toth et al. [29] argue that parametric dependencies constructed through integrated dynamic couplings will not only provide integration with the model domain, but also will support the development of more integrated design practices that facilitate a shared understanding and knowledge between disciplines.

Distributed models, such as integrated dynamic models and platforms like VDS [19], are superior to combined model approaches in terms of BPS choices. The growing number of dedicated BPS plugins built for design tools are making central models almost as flexible in terms of BPS choices as the distributed models. The flexibility in central models, however, is rather limited, if the operators need to couple various calculation features from one BPS to another. The presence of middleware in the distributed model is in some cases able to couple multiple BPS and give a combined feedback to the operators, which is obviously not possible with only one BPS tool alone. Jakubiec and Reinhart [75] showed a coupling of daylight simulations (DAYSIM) and thermal simulations (Energy+) providing rapid feedback from within existing design models (created with Rhino). The results are now published as a middleware plugin (DIVA) supporting Grasshopper. DIVA has in turn been partially integrated into other BPS couplings e.g. Honeybee [77] with (RADIANCE and OpenStudio [82]), however the real benefit of using the integrated dynamic models is the option to couple various BPS by operator choice, and not only by what is made possible by developers. The combined and central model does not support the same operator-to-model freedom.

Design tools such as Rhino, Revit and SketchUp are blurring the distinction of being a pure CAD-program, as these design tools support increasingly many BPS environments either as services, or as plugins. While the plugins offer extension capabilities beyond the design tool, the design tool itself requires an exclusive operator to use the extensions. In this regard the plugins and services can either be characterized as middleware elements supporting the distributed model method, but controlled in the design tool interface, or in the situation of low level runtime couplings as a combined calculation model. Nevertheless, most design tools support one or more scripting environments and are capable of utilizing a VPL as middleware, why integrated dynamic models are gradually becoming a prevailing method of integrating design tools with BPS tools in practice.

4.2. Ontology challenges

One of the main barriers of translating data from a design tool into input a BPS tool can understand is adding sufficient non-geometric and correct properties to the geometry. An example is the apparently simple task of drawing a window in a design tool. The designer draws a square surface which is clearly representing a window for a human, but for a BPS environment it solely represents a surface with zero properties. It is not even near to being understood as window, even when the designer has drawn mullions and transoms and hinges. This kind of problem is often called an ontology challenge.

In the past decade the introduction of IFC and other standards like gbXML have sought to solve these types of challenges. An object is now not only representing a three dimensional shape but can hold different properties such as price, and heat capacity. What is equally important is the object to object relationship, such as the fact that a wall can contain a window, but not the other way around. To handle such relationships semantic rules have been introduced.

Semantic rules work as natural language concepts like a wall cannot be inside a window [83]. While the inclusion of semantic rules is aiding a contextual human understanding of the model, it is also providing interoperability improvements between the design tool and the BPS environment. One of the main concepts of semantic rule translation is the derivation of properties of topological and geometrical data from a semantic construct. Semantic constructs often rely on databases [47] and differ from usual data models as additional semantic information is included, e.g. classes and properties. Rezgui [84], referring to the implementation of OWL, classifies semantic constructs as information that helps defining relationships of objects. Semantic rule constructed through ontology-based approaches is of no means a cure-all for the illnesses facing ontology-challenges, but it represents some of the work being done in adding intelligence into integrated models. Nonetheless bidirectional semantic rules would lead to better model management and relevant feedback to the designer, as suggested by Duan et al. [83].

Regardless of the use of semantic rules or conditional programming (such as if-then(-else)), basic constructs of logics have to be applied to any generic system, if the model is to make use of linked BPS. Evidently all BPS environments dictate a strict system of constructs in reading, running and evaluating a given geometric model (often denoted as program specific syntax), and therefore the generic model must conform to the limitations and rules set by the linked BPS.

The ontology challenges are one of the main reasons, why combined and central models do not work across multiple design tools and multiple BPS. Both the combined and central models pursue to use building schemes/languages such as IFC as main import option, thus standardizing semantic rules and minimize ontology challenges. Supporting open standards is seemingly the best way to cover multiple design tools, but many BPS tools still do not support the full compliance of e.g. IFC. Belsky and Sacks [85] suggest a framework of enriching IFC files with an automated approach, deriving semantically useful geometric model data from the explicit and implicit information contained in an IFC schema. Essentially such framework could be developed in delivering enriched IFC files to a compatible BPS tool, however this is yet to be seen.

Distributed models do not rely on one ontology standard, instead different operator defined semantic rules may be defined from project to project, as the VPL provides a highly flexible ontology framework. Davis and Peters [64] discuss the benefit of having content-based relationships on geometrical objects, but the truth is that the VPL does not differ between geometric and non-geometric semantic rules, why it is superior to half implemented ontology standards. Pratt et al. [86] and Toth [87] have suggested to use a common protocol between design tools and BPS instead of a common format to formalize the use design tool couplings to BPS. Essentially the protocol is a middleware aiding the operators in creating fast coherent conceding calculation models inside the design tool. While Pratt et al. describe a system able to support various design tools with and without VPLs, Toth relies on the VPL as the middleware and is thereby essentially describing an integrated dynamic model.

4.3. Model scalability

While we have been discussing who and how, it might be relevant to ask when. When should the BPS be introduced into the model in the early design process? Shaviv et al. [88] suggest using a gradual progressing result as a combination of two evaluation approaches, as seen in Fig. 11. The procedure begins with intuition in a heuristic-knowledge based system. As the design process progresses, so does the use of calculation models progresses into more precise simulation-based systems. Banke [6] argues that architects perform mental simulations to imagine what-if scenarios and as a result, the design process is highly affected by intuition and tacit knowledge, where rules of thumb instead of actual calculations are used from the earliest design stage.

In an interview with practitioners and building physicists by Hopfe et al. [21], the question of when BPS tools should be introduced was addressed. All interviewees seem to agree that current focus on building performance has to be accounted for earlier in the process and that consultants and specialists were rarely invited into the early design process, resulting in less integrated and insufficient design solutions. The use of BPS over heuristic knowledge divided the interviewees, yet the interviewees stated that future tools should address a multitude of value drivers, should be easy to use, be able to represent complex scientific phenomena, and that they should be tested and validated. As all interviewees agreed that experience is essential for developing design concepts, it may be difficult to see, how tools can be used without specialist attending the early design stages, and it is even more difficult to see, how purely heuristic models can accommodate every aspect of a future tool requirement.

Struck [81] argues that tools for the early design stages must be flexible enough to facilitate expanding representations with innovative design concepts, thus being able to dynamically scale the model resolution to fit the different levels of information density. Attia et al. identified the need of tools/environments that allow the gradual support of BPS feedback: Users need fluid tools that can produce initial results from a rough building representation during early design phases and at the same time allow for detailing of building components during later design phases [22]. Most BPS tools do not support such a large range of representational variations, why different tools is needed in different stages. Struck [81] categorizes the potential of using different BPS tools in two possible stages (conceptual and detailed design stages). While simplified tools might prove to be useful at a certain point in the early design, they might be too limited to be applied in later design evaluations. Support of multiple design tools is reserved the central and distributed model method. The combined model may by the least supportive method when seeking to integrate BPSs gradually into the design process simply because of the limitations of being bound to one or very few BPS tools. The central model method and the distributed model method support a wider range of BPS tools and comply with the procedural design stages required of the design process.

4.4. Model speed

Various approaches are proposed on informing architects in the early progressing stages of design and some are faster than others. Some methods e.g. the prototype approach [7] seeks on delivery of instantaneous performance result, and thereby bringing that information into the creative process of designing a building. The prototype approach is highly heuristic and often relies on user defined and simplified rules-of-thumb calculations rather than linking to a BPS tool, much in line with Banke's [6] arguments when talking about the earliest explorations of building design.

When it comes to calculation speed, the calculation model in the BPS environments is the main limiting factor. Precision and speed are inversely related why some BPS tools may be faster than others, but less precise. Precision of any controlled calculation is also related to the quality of data given within the system. The quality of the model is vital for the quality of any simulation or calculation result. In the early design stages the amount of data available in the model is limited, since no definite decisions are defined. Later in the process, as the model grows in detail and so does the data space necessary for an increasingly more precise result. Consequently different stages of the early design have different limits to the precision of any calculations and simulations. Hensen [35] notes that using a correct simulation methodology as well as the appropriate level of modeling resolution is often underestimated. Hensen further argues that lower resolution (faster and less precise) methods would be quite sufficient and much more efficient. CFD (computational fluid dynamics) or more computational heavy simulations (e.g. ray tracing) could seem “over the top”, when sufficient precision can be modeled with simpler methods. In the end all that matters is solving the right equations sufficiently accurate, as opposed to solving the wrong equations right. The balance is therefore improving the precision of the results and getting the result as fast as possible. A balance is either biasing towards model responsiveness if asking an architect or towards precision if asking an engineer [22].

4.5. Model precision and validity

While a combined model provided by commercial products like IESVE is very likely to be stable, tested and based on validated BPS results, the central and distributed models does not have the same privileges. Operators must be aware of the consequences of coupling design tools to un-validated BPS tools.

In terms of precision linked BPS multi-zone environments provides both validated and precise results, whereas simpler simulation single-zone tools (such as DOE-2 [89], Be10 [90]) do a sufficient job in terms of precision in the early design stages, when less information is available. Over the past few years many different researches have published various methods containing high precision calculations, some approaches focuses on manual variations (e.g. [6,8,91]), while others utilizes sensitivity analyses with Monte Carlo algorithms to inform the designer (e.g. [10,92,93]). Most approaches including those that rely on optimization algorithms [27,69,94–98] focus on optimizing singular or very few objectives (e.g. yearly energy consumption). In general

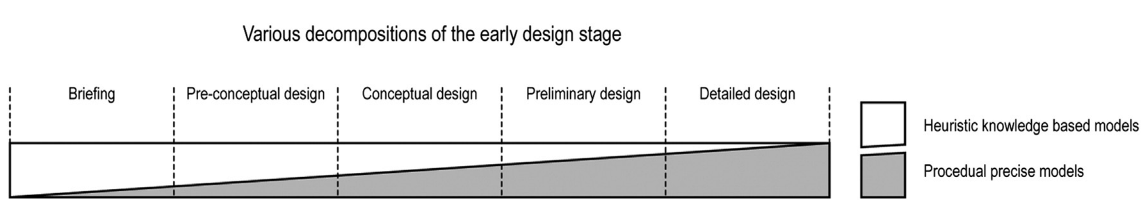


Fig. 11. Transition of heuristic knowledge based models into procedural precise models over different design stages. Based on Shaviv et al. [88].

such methods seek to maintain high precision of performance evaluation, which in turn compromises the speed of delivering results. Non-commercial and non-combined models can obviously support validated BPS tools, but they can also support the opposite. The problem may be most pertinent when discussing the integrated dynamic models because of the opportunity of using a VPL as a BPS tool.

User defined scripts that evaluate building performance and un-validated BPS tools are another way of estimating performance. The main reason user-scripted tools and un-validated BPS are being used, is the potential of a much faster integrated dynamic model, which potentially generates results instantaneously. In a small survey performed by Banke [6], the speed of daylight performance feedback was more important in the early stages than the later design stages. Architects, in contrast to engineers, may more often resort to un-validated BPS tools as only 18% ranked validated performance results as their first preference, while engineers ranked validated results at 29% [22]. Obviously such non-conventional tools are difficult to validate. Ideally only validated BPS tools should be linked with minimum latency and provide high level of customization options for the users.

4.6. Usability and customization

BPS tools that provide a friendly graphical user interface GUI was ranked higher between architects and engineers than those with a more complex GUI [22]. Usability is because of this a very important factor when choosing tools that contain a model for the early design stages. On the other hand, the features and flexibility of the tools are paramount to support the nuances that the operators' analyses need. Therefore, the model method must not sacrifice features and flexibility with e.g. a simplistic tool. User customization is partially a matter of being able to link different functionalities from different BPS tools and partially related to modeling options to create a design, why customization, unlike usability, can be tool-independent.

The combined model is able to provide a high level of usability while supporting little or no customization. The central model provides high levels of customization as well as feature, and depended on the linked software tools the usability of the central model can vary. The highest level of customization is provided by the distributed model, since the middleware can be customized to do everything from generating solutions to suggesting, which BPS to use. Frameworks such as VDS [19] may also provide the same good usability, as commercial products deliver with combined models.

Toth [87] suggests collaborative use of integrated dynamic models, where different disciplines are able to work in parallel. While same collaborative ideas are found in other distributed models (e.g. VDS), the ability for the user to customize the design environment to fit the special needs of the particular operators is an exclusive characteristic of the integrated dynamic model. Even though Amor and Faraj [42] identified object-oriented modeling (e.g. C++, Java) as the obvious solution to many of the problems associated with central models, middleware operated in low-level programming languages such as C++ is difficult for non-programmers to access. The fact that low-level programming expertise is relatively uncommon among architects and engineers [99] makes the lack of support for non-programmers in creating custom software connections in combined and central model methods very unlikely.

The presence of a VPL in an integrated dynamic model may be the most important factor, when user customization is valued high in the early design stage. Integrated dynamic models, in contrast to simulation packages and using conventional IFC og gbXML formats, provides the capacity of scripting. VPLs such as Grasshopper even provide distribution frameworks for customized user scripts. Open forums, free and commercial plugins encourage non-programmers to customize the model and the model environment, which is not seen in either combined models or central models.

4.7. Automation and design decision support

While the interoperability between design tools and BPS and ontology rules is covered by automation as a term, the automation in generating geometry, generating alternative solutions, and automating decisioning is increasingly used by building designers and simulationists. Automation and design support beyond the capabilities of a design tool and a BPS tool is very likely to become standard in practice. Many of the automation techniques seek to provide better design support, than can be derived from building performance simulations alone. Design support ranges from generating sensitivity feedback, e.g. Struck [81], to support in multi criteria decision making, e.g. Gerber and Lin [100]. A common characteristic of these types of automations is that they are highly depended on the specific building design and that every model needs a custom automation procedure. Essentially these (in some cases highly advanced) automation features are today largely restricted to the distributed model method, where one (or more) middleware is the source of automation, and simulation packages, e.g. BEopt [101] and iDbuild [8], incorporate automation and design support in combined models.

While other papers e.g. Attia et al. [102], have identified current trends and future possibilities of design support utilizing combined models, the similar overview of integrated dynamic models has not been reviewed. Of this reason some of the key features provided by a VPL are discussed here.

Generating geometry through geometric automation,[5] often referred as scripting, is becoming increasingly popular with the inclusion VPL. Architects are increasingly using VPL and other scripting environments that defines the parametric method of design exploration. Kotnik [103] has richly described the differences between the methods; the generative, parametric and algorithmic approaches. A common characteristic of the methods is their ability to automate generation of advanced geometric compositions through either a VPL or another high level programming language. That is, why VPLs can be powerful tools that increase the creativity and productivity of building designers [104].

Rules liberate according to March [105] and as Chase [106] elaborates: generative design tools introduce key concepts that are the foundations of design, e.g., geometry, spatial relations and transformations, recursion, reiteration, procedures, and encapsulation. When it comes to automating geometry generation, different ways of shape control have been introduced over the years. Early pioneering research such as C. Alexander's Pattern Languages [107] are based on scientific reasoning and seeks to define system thinking to architecture. Many other sources from mathematics, physics and nature have over the years inspired architects in controlling and modifying their design. Alexander argues that the most quality-defined attributes can be translated into computational algorithms through pattern finding processes. Stiny [108] later defined the shape grammar paradigm (Fig. 12), which is basically a set of rules of transformation, applied recursively to an initial geometry, generating new forms.

Caldas [109] showed different variations of an architecturally defined approach to generative methods involving a BPS (DOE2.1E). Caldas used the VPL to create different shape grammar functions and linking the parametric geometric model to a calculation model, which essentially is an integrated dynamic model employing geometric automation. Integrated dynamic models employing shape grammar have recently been used in combination with various genetic optimization algorithms. Many of these models are defined with objective functions that are building performance related as suggested by Duarte [110] and Byrne et al. [111]. Shape grammar has also been suggested in architecture as a more formal design method exemplified by the design constraint systems suggested by Bollmann and Bonfiglio [112]. Shape

[5] Geometric automation is here associated with simple geometry-generating tasks such as creating a plane to more advance techniques generating mathematically defined free form geometry usually difficult to create through traditional CAD interfaces.

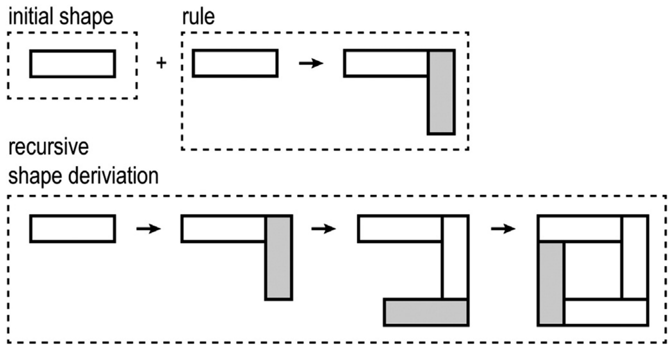


Fig. 12. Example of shape grammar. Generation of form based on simple rules based on [108].

grammar and other geometrically defined automation algorithms are able to generate form that in traditional (manual) ways would be very difficult to achieve. But the most important influence of using geometrical automation is the option to evaluate numerous variations of a geometrical concept extremely fast.

> Rather than the designer creating the design solution as in conventional design tool, the idea is that the designer first establishes the relationships by which parts connect, builds up a design using these relationships and modifies the relationships by observing and selecting from the results produced. The system takes care of the job of keeping the design consistent with the relationships and thus increases designer ability to explore ideas by reducing the tedium of rework.
>
> [Woodbury [113]]

Current implementations of building information formats such as IFC lack support for parametric and dynamic modeling the early design stages. Steel et al. [114] clarify on the subjects of interoperability and dynamic data exchange: Presently, models that incorporate parameterization can only be exported into IFC by fixing the parameters and exporting a snapshot of the design. This is obviously less than ideal, particularly when exchanging models early in the design process. Steel et al. even argue that the IFC schema is fundamentally incapable to support parametric modeling: incorporating support for parametric modeling into the IFC language would involve a significant reworking of the IFC standard [114]. In other words, it is at present time difficult to utilize the IFC BIM-framework as a foundation for dynamic design explorations, let alone performing geometric automation such as shape grammar with common schemas. However, in relation to the distributed model method some researchers e.g. Holzer [66] on DesignLink [115] describe methods using features similar to shape grammar, where gbXML is a part of larger system facilitating and automating data transfer between the design tool and BPS environment.

At present the most direct way to implement parametric relationships and geometric automation is utilizing the integrated dynamic model method.

The true potential of evaluating performance (e.g. annual building energy consumption) of multiple solutions is only present, when the geometry itself is fully parametric. This means that the building designer, or whoever is operator of the geometric model, has to establish geometrical relationships between every object that might be affected by changes in the model. The same relationships must be defined for the non-geometrical objects affecting changes in the BPS. This true potential has been explored by various researchers, e.g. Duarte [110] who takes the fully parametric integrated dynamic model and uses optimization algorithms to search and optimize on the performance feedback provided by the BPS. Duarte prefers weighing averages of the objectives and afterwards fine tuning the synthetic goal seeking of the shape grammars. Other researchers, e.g. Caldas [109] suggest using a penalty system in conjunction with dependency variables to adjust fitness weight on a geometric evaluation. Caldas argues that the implication of introducing dynamic constraints as penalty functions on shape grammar gives the building designer (model operator) more direct control of the geometry when using genetic (or other optimization) algorithms.

It is debatable, whether shape grammar and generative rules are in conflict with the architectural design exploration. One of the main arguments against automation of geometry in designing buildings is the fact that decisions ordinarily performed by a human designer is by automation held and controlled by a computer, which obviously eliminates the function of the building designer. Killian [15] argues that grammatical approaches fail in the context of open design problems due to their reliance on generative rules, partially based on the limitations of the hierarchical structure of the dependency chains that are present in rule based definitions. One can argue that the design problem of architecture is never fully open, since design is confined to fundamental constraints of human needs, of government imposed legislations, economy as well as performance-related criteria. And then again, referring to Davis and Peters [64], the worst case is when the limitations of the model environment dissuade the operator to make the right choices. The operator must of this reason be very observant when utilizing automation such as shape grammar. Additional discussions related to utilizing computers to automate design in architecture can be found in a recent review on the subject by Grobman et al. [116].

What is important to notice is that generative rules such as shape grammar are required, if the model needs to support any autonomous optimization procedures. Without the parametric dependencies between objects (ontology) and rules that govern, in which direction the geometry is allowed, optimization of building performance is very unlikely to be used in practice. The only model method that supports shape grammar and operator defined semantics is the integrated dynamic model.

5. Conclusion

In the process to support the design of better performing buildings, the current research in the use of building performance simulations (BPSs) in the early design stages has been explored. Strictly, concerning model level operation and model level collaboration, the focus of this review is to answer two main questions:

1) Who operates geometric models and building calculation models?
2) What is the best way to couple geometric models and building calculation models in the early design stages?

The review consists of two parts to cover the main aspects of the questions. The conclusions of both questions are in many ways overlapping, as the domains of model interaction and human interaction are undeniably interrelated.

1) As building performance, such as energy performance is a quantity defined performance, energy BPS and other calculable performances are relatively straightforward to compare and optimize. Current research has shown that regardless of collaborative arrangements, building performance with the aid of BPS, is likely to be improve in the early design stages. Most studies suggest that mixed design teams, consisting of both engineers and architects, representing the expertise within their respective disciplinary fields, will achieve the best performance outcome.
2) In the effort of designing better performing and healthier buildings with the aid of design tools, scripting, and building performance simulations, several advancements have been published during the past few years. Integrated dynamic models, such as the combination of a design tool, a visual programming language, and a BPS tool provide better support in terms of BPS tool diversity, flexibility of feedback and multidisciplinary collaboration during the early design stages. This is opposed to alternatives such as the current implementation

of IFC or gbXML, or the unaccompanied use of a simulation packages. However, integrated dynamic models lack the convenience of standardized formats or common interoperability methods. Therefore, making a design tool and a BPS tool work truly efficient together creates unique interoperability challenges every single time a new model is constructed.

Social performance and similar quality defined performances rely on human evaluations. At present time, there are no simple way to jointly combine, compare or optimize, quality- and quantity defined performances. Nevertheless, the conclusion is that the highest potential of creating better performing buildings while respecting architectural integrity is by utilizing optimization techniques and other automations in combination with an integrated dynamic model. Integrated dynamic models combined with optimization algorithms have proved to provide a potent method of evaluating multiple solutions. Further, these models have the ability to provide fast, nearly instantaneous, feedback to the designer or to focus on precise and elaborate multivariate analyses. However, the use of optimization comes at a cost; the building designer must accept the process of defining clear goals and rely on shape grammar or similar techniques to control the direction of the design.

Acknowledgments

The work is part of PhD research currently held at the Technical University of Denmark, and is performed in collaboration with Grontmij and Innovationsfonden.

References

[1] A. Hermund, Anvendt 3D modellering og parametrisk formgivning, Det Kongelige Danske Kunstakademis Skoler for Arkitektur, Design og Konservering, 2012.
[2] IFC — BuildingSMART, 2013.
[3] C. Bleil de Souza, Contrasting paradigms of design thinking: the building thermal simulation tool user vs. the building designer, Autom. Constr. 22 (2012) 112–122.
[4] gbXML.org. Open Green Building XML gbXML, 2014.
[5] S. Attia, A Tool for Design Decision Making, Université catholique de louvain, 2012.
[6] T. Banke, Parametri i praksis — Generativ performance i arkitektur, Det Kongelige Danske Kunstakademis Skoler for Arkitektur, Design og Konservering, 2013.
[7] J. Klitgaard, P.H. Kirkegaard, M. Mullins, On the integration of digital design and analysis tools, in: A. Ali, C.A. Brebbia (Eds.),Digit. Archit. Constr, 2006, pp. 187–196 (Seoul).
[8] S. Petersen, Simulation-based Support for Integrated Design of New Low-energy Office Buildings, Technical University of Denmark, 2011.
[9] R. Mora, C. Bédard, H. Rivard, A geometric modelling framework for conceptual structural design from early digital architectural models, Adv. Eng. Inform. 22 (2008) 254–270.
[10] C. Struck, P.J.C.J. de Wilde, C.J. Hopfe, J.L.M. Hensen, An investigation of the option space in conceptual building design for advanced building simulation, Adv. Eng. Inform. 23 (2009) 386–395.
[11] M.K. Holst, P.H. Kirkegaard, Computational design tools for integrated design, ICSA 2010 — 1st Int. Conf. Struct. Archit. ed. Cruz, ISBN: 978-0-415-49249-2, 2010, pp. 471–472.
[12] P. Kataras, Multi-objective Performance Optimization of Building Form Based on the Principle of Analytical Geometry, University College London, 2010.
[13] B. Welle, J. Haymaker, Z. Rogers, ThermalOpt: a methodology for automated BIM-based multidisciplinary thermal simulation for use in optimization environments, Build. Simul. 4 (2011) 293–313.
[14] X. Shi, Design optimization of insulation usage and space conditioning load using energy simulation and genetic algorithm, Energy 36 (2011) 1659–1667.
[15] A. Kilian, Design Exploration Through Bidirectional Modeling of Constraints, Massachusetts Institute of Technology, 2006.
[16] G. Löhnert, A. Dalkowski, W. Sutter, Integrated Design Process Task 23 Subtask B Design Process Guidelines, Solar Heating and Cooling Programme, IEA, 2003.
[17] D. Holzer, BIM's seven deadly sins, Int. J. Archit. Comput. 09 (2011) 463–480.
[18] J.R. Haymaker, MACDADI, http://www.macdadi.com 2011.
[19] P. Michael Pelken, J. Zhang, Y. Chen, D.J. Rice, Z. Meng, S. Semahegn, et al., "Virtual Design Studio"—part 1: interdisciplinary design processes, Build. Simul. 6 (2013) 235–251.
[20] J. Byrne, M. Fenton, E. Hemberg, J. McDermott, E. Shotton, C. Nally, Combining structural analysis and multi-objective criteria for evolutionary architectural design, Appl. Evol. Comput. 204–213 (2011).
[21] C. Hopfe, C. Struck, G. Harputlugil, J. Hensen, P. De Wilde, Exploration of the use of building performance simulation for conceptual design, Proc. IBPSA-NVL Conf, 2005, pp. 1–8.
[22] S. Attia, J.L.M. Hensen, L. Beltrán, A. De Herde, Selection criteria for building performance simulation tools: contrasting architects' and engineers' needs, J. Build. Perform. Simul. 5 (2012) 155–169.
[23] Henning Larsen Architects, Henning Larsen Architects, http://www.henninglarsen.com/architecture/sustaina 2012.
[24] Alectia, Alectia, http://www.alectia.com/index.php?cID=237 2012.
[25] V. Soebarto, Teaching an energy simulation program in an architecture school: lessons learned, Proc. Ninth Int. IBPSA, 2005, pp. 1147–1154.
[26] Y.E. Kalay, Performance-based design, Autom. Constr. 8 (1999) 395–409.
[27] X. Shi, W. Yang, Performance-driven architectural design and optimization technique from a perspective of architects, Autom. Constr. 32 (2013) 125–135.
[28] A. Pasold, I.W. Foged, Function follows performance in evolutionary computational processing-vertical evolution, Int. J. Archit. Comput. 8 (2010) 525–536.
[29] B. Toth, F. Salim, J. Burry, J. Frazer, R. Drogemuller, M. Burry, Energy-oriented design tools for collaboration in the cloud, Int. J. Archit. Comput. 09 (2011) 339–360.
[30] C.E. Ochoa, M.B.C. Aries, J.L.M. Hensen, State of the art in lighting simulation for building science: a literature review, J. Build. Perform. Simul. 5 (2012) 209–233.
[31] P. Sanguinetti, S. Abdelmohsen, J. Lee, J. Lee, H. Sheward, C. Eastman, General system architecture for BIM: an integrated approach for design and analysis, Adv. Eng. Inform. 26 (2012) 317–333.
[32] N. Thuesen, P.H. Kirkegaard, R.L. Jensen, Evaluation of BIM and Ecotect for conceptual architectural design analysis, in: W. Tizani (Ed.)Proc. Int. Conf. Comput. Civ. Build. Eng. Icccbe, 2010.
[33] A. Zarzycki, Exploring parametric BIM as a conceptual tool for design and building technology teaching, Proc. SpringSim'10, ACM Press, New York, ISBN: 978-1-4503-0069-8, 2010.
[34] Integrated Environmental Solutions, IESVE, http://www.iesve.com/ 2013.
[35] J.L.M. Hensen, Towards more effective use of building performance simulation in design, Proc. 7th Int. Conf. Des. Decis. Support Syst. Arch. Urban Planin, 2004, pp. 2–5.
[36] BuildingSmart.org. BuildingSMART, 2013.
[37] A. Gupta, Developing a BIM-based Methodology to Support Renewable Energy Assessment of Buildings, Cardiff University, 2013.
[38] A. Gupta, A. Cemesova, C. Hopfe, Y. Rezgui, T. Sweet, Development of a solar PV simulation tool compatible with industry foundation classes, 13th Conf. Int. Build. Perform. Simul. Assoc., Chambery, 2013.
[39] LBNL Simergy, Lawrence Berkeley Natl. Lab, 2014.
[40] NREL U.S. Department of Energy, OpenStudio, 2014.
[41] C.M. Rose, V. Bazjanac, An algorithm to generate space boundaries for building energy simulation, Eng. Comput. 31 (2) (2013) 271–280.
[42] R. Amor, I. Faraj, Misconceptions about integrated project databases, J. Inf. Technol. Constr. 6 (2001) 57–68.
[43] S. Moosberger, Application of specific models in a general simulation tool, Build. Simulation, Elev. Int. IBPSA Conf, 2009, pp. 1603–1608.
[44] Equa, IDA Indoor Climate and Energy IDA Indoor Climate and Energy simulation software, http://www.equa-Solutions.co.uk/en/software/idaice 2014.
[45] J. Plume, J. Mitchell, Collaborative design using a shared IFC building model—learning from experience, Autom. Constr. 16 (2007) 28–36.
[46] B. Toth, P. Janssen, R. Stouffs, A. Chaszar, S. Boeykens, Custom digital workflows: a new framework for design analysis integration, Int. J. Archit. Comput. 10 (2012) 481–500.
[47] Y.E. Kalay, Enhancing multi-disciplinary collaboration through semantically rich representation, Autom. Constr. 10 (2001) 741–755.
[48] G. Wurzer, Schematic systems-constraining functions through processes (and vice versa), Int. J. Archit. Comput. 8 (2010) 201–217.
[49] S. Van Nederveen, E. Bektas, B. Luiten, H.M. Bõhms, Distributed modeling for road authorities, Proc. 30th CIB W78 Int. Conf., Beijing, 2013, pp. 642–651.
[50] V. Bazjanac, Building energy performance simulation as part of interoperable software environments, Build. Environ. 39 (2004) 879–883.
[51] D. Greenberg, K. Pratt, B. Hencey, N. Jones, L. Schumann, J. Dobbs, et al., Sustain: an experimental test bed for building energy simulation, Energy Build. 58 (2013) 44–57.
[52] J. Zhang, P. Michael Pelken, Y. Chen, D.J. Rice, Z. Meng, S. Semahegn, et al., "Virtual Design Studio"—part 2: introduction to overall and software framework, Build. Simul. 6 (2013) 253–268.
[53] Autodesk, Revit, http://www.autodesk.dk/revit 2013.
[54] Robert McNeel, et al., Rhino, http://www.rhino3d.com 2013.
[55] Trimble, SketchUp, http://www.sketchup.com 2013.
[56] CHAMPS MultiZone building application, http://champs.syr.edu/software/champs_mz/champs_mz.html 2013.
[57] U.S. Department of Energy, Energy+, http://apps1.eere.energy.gov/buildings/energyplus/ 2013.
[58] Robert McNeel, et al., Grasshopper3d, http://www.grasshopper3d.com 2013.
[59] Autodesk, Dynamo, http://wikihelp.autodesk.com/Vasari/enu/Community/ 2013.
[60] Bentley, GenerativeComponents, http://www.bentley.com/GetGC 2013.
[61] Gehry Technologies, Digital Project, http://www.gehrytechnologies.com/digital–project 2013.
[62] D. Davis, Yeti3d, http://www.yeti3d.com 2013.
[63] Bentley, MicroStation V8i, http://www.bentley.com/en-US/Products/MicroStation/ 2014 (2014).
[64] D. Davis, B. Peters, Design ecosystems: customising the architectural design environment with software plug-ins, Archit. Des. 83 (2013) 124–131.
[65] D. Holzer, S. Downing, Y. Tengono, Delivering Digital Architecture in Australia, DDAA Presentation, Australian Research Council, 2009.
[66] D. Holzer, Optioneering in collaborative design practice, Int. J. Archit. Comput. 8 (2010) 165–182.

[67] Autodesk, Ecotect, http://usa.autodesk.com/ecotect–analysis/ 2013.
[68] S. Bambardekar, U. Poerschke, The architect as performer of energy simulation in the early design stage, Proc. 11 Th Int. IBPSA Conf. Glas. Scotl, 2009, pp. 1306–1313.
[69] S.E. Lin, D.J. Gerber, W. Hall, L. Angeles, Designing — in performance: towards cloud based simulation and multidisciplinary design solution space search, SimAUD 2013 Symp. Simul. Archit. Urban Des, 2013.
[70] Autodesk, Vasari, http://autodeskvasari.com/VasariBeta3 2013.
[71] DIVA/Viper, http://www.food4rhino.com/project/diva-Rhino-0. 6 2013.
[72] ArchSim Energy Modeling, http://www.food4rhino.com/project/archsim-Em.9 2014.
[73] Transient System Simulation Tool TRNSYS, http://www.trnsys.com/ 2013.
[74] Geco, http://www.food4rhino.com/project/geco 2013.
[75] J.A. Jakubiec, C.F. Reinhart, DIVA 2.0: integrating daylight and thermal simulations using Rhinosoros 3D, DAYSIM and ENERGYPLUS, 12th Conf. Int. Build. Perform. Simul. Assoc, 2011, pp. 14–16.
[76] Christoph Reinhart, Daysim, http://daysim.ning.com/ 2013.
[77] M. Rousdary, C. Mackey, Honeybee, http://core.thorntontomasetti.com/honeybee-Is-out-and-Flying/ 2014.
[78] D. Davis, Modelled on Software Engineering: Flexible Parametric Models in the Practice of Architecture, RMIT University, 2013.
[79] M. Burry, Scripting Cultures: Architectural Design and Programming, AD Primers, 2013.
[80] S. Citherlet, Towards the Holistic Assessment of Building Performance Based on an Integrated Simulation Approach, Swiss Federal Institute of Technology, 2001.
[81] C. Struck, Uncertainty Propagation and Sensitivity Analysis Techniques in Building Performance Simulation to Support Conceptual Building and System Design, Eindhoven University of Technology, 2012.
[82] E. Hale, D. Macumber, K. Benne, D. Goldwasser, Scripted building energy modeling and analysis, SimBuild 5th Natl. Conf. IBPSA, 2012.
[83] Y. Duan, C. Cruz, C. Nicolle, Managing semantics knowledge for 3D architectural reconstruction of building objects, Eighth ACIS Int. Conf. Softw. Eng. Res. Manag. Appl, ISBN: 978-0-7695-4075-7, 2010, pp. 11–18.
[84] Y. Rezgui, S. Boddy, M. Wetherill, G. Cooper, Past, present and future of information and knowledge sharing in the construction industry: towards semantic service-based e-construction? Comput. Des. 43 (2011) 502–515.
[85] M. Belsky, R. Sacks, A framework for semantic enrichment of IFC building models, Proc. 30th CIB W78 Int. Conf., Beijing, 2013, pp. 514–523.
[86] K. Pratt, N. Jones, L. Schumann, A.D. Heumann, Automated translation of architectural models for energy simulation, SimAud '12, San Diego, ISBN: 978-1-61839-790-4, 2012, pp. 65–72.
[87] B. Toth, Closing the loop of design and analysis, CAADRIA 16th Int. Conf. Comput. Archit. Des. Res, 2011, pp. 525–534.
[88] E. Shaviv, Y.E. Kalay, U.J. Peleg, An integrated knowledge-based and procedural system for the design of energy conscious buildings, Autom. Constr. 1 (1992) 123–141.
[89] James J. Hirsch, et al., DOE-2, http://www.doe2.com/ 2013.
[90] SBI, Be10, http://www.be10.sbi.dk/ 2013.
[91] H. Poirazis, Å. Blomsterberg, Energy and thermal analysis of glazed office buildings using a dynamic energy simulation tool, Ninth Int. IBPSA Conf, 2005.
[92] C.J. Hopfe, Uncertainty and Sensitivity Analysis in Building Performance Simulation for Decision Support and Design Optimization, Technische Universiteit Eindhoven, 2009.
[93] C.J. Hopfe, J.L.M. Hensen, Uncertainty analysis in building performance simulation for design support, Energy Build. 43 (2011) 2798–2805.
[94] Á. Fialho, Y. Hamadi, M. Schoenauer, Optimizing architectural and structural aspects of buildings towards higher energy efficiency, Proc. 13th Annu. Conf. Companion Genet. Evol. Comput. — GECCO'11, Dublin, 2011, p. 727.
[95] W. Wang, H. Rivard, R. Zmeureanu, An object-oriented framework for simulation-based green building design optimization with genetic algorithms, Adv. Eng. Inform. 19 (2005) 5–23.
[96] A. Sharag-Eldin, A parametric model for predicting wind-induced pressures on low-rise vertical surfaces in shielded environments, Sol. Energy 81 (2007) 52–61.
[97] B. Raphael, Multi-criteria decision making for collaborative design optimization of buildings, Built Environ. Proj. Asset Manag. 1 (2011) 122–136.
[98] F.Ø. Pedersen, A Method for Optimizing the Performance of Buildings, Technical University of Denmark, 2006.
[99] F. Flager, B. Welle, P. Bansal, Multidisciplinary process integration and design optimization of a classroom building, J. Inf. Technol. Constr. 14 (2009) 595–612.
[100] D.J. Gerber, S.E. Lin, Synthesizing design performance: an evolutionary approach to multidisciplinary, ACADIA 2012 Synth. Digit. Ecol, 2012, pp. 66–75.
[101] C. Christensen, S. Horowitz, T. Givler, A. Courtney, G. Barker, BEopt: software for identifying optimal building designs on the path to zero net energy preprint, ISES 2005 Sol. World Congr, 2005.
[102] S. Attia, E. Gratia, J.L.M. Hensen, Simulation-based decision support tool for early stages of zero-energy building design, Energy Build. 49 (2012) 2–15.
[103] T. Kotnik, Digital architectural design as exploration of computable functions, Int. J. Archit. Comput. 8 (2010) 1–16.
[104] C. Eckert, I.A.N. Kelly, M. Stacey, Interactive generative systems for conceptual design: an empirical perspective, AI EDAM, 132012. 303–320.
[105] L. March, Rule bound unruliness, Environ. Plan. B, Plan. Des. 23 (1996) 391.
[106] S.C. Chase, Generative design tools for novice designers: issues for selection, Autom. Constr. 14 (2005) 689–698.
[107] Christopher Alexander, S. Ishikawa, M. Silverstein, A Pattern Language: Towns, Buildings, Constructions, 1974.
[108] G. Stiny, Introduction to shape and shape grammars, Environ. Plan. B, Plan. Des. 7 (1980) 343–351.
[109] L. Caldas, An Evolution-based Generative Design System: Using Adaptation to Shape Architectural Form, Massachusetts Institute of Technology, 2001.
[110] J.P. Duarte, Customizing Mass Housing: A Discursive Grammar for Siza's Malagueira Houses, Massachusetts Institute of Technology, 2001.
[111] J. Byrne, E. Hemberg, M. O'Neill, Interactive operators for evolutionary architectural design, Proc. 13th Annu. Conf. Companion Genet. Evol. Comput. — GECCO'11, Dublin, 2011, p. 43.
[112] D. Bollmann, A. Bonfiglio, Design constraint systems — a generative approach to architecture, Int. J. Archit. Comput. 11 (2013).
[113] R. Woodbury, Elements of Parametric Design, Routledge, 2010.
[114] J. Steel, R. Drogemuller, B. Toth, Model interoperability in building information modelling, Softw. Syst. Model. 11 (2010) 99–109.
[115] Arup, DesignLink, http://arupforge.arup.com/wiki/index.php?title=Des 2013.
[116] Y.J. Grobman, A. Yezioro, I.G. Capeluto, Computer-based form generation in architectural design — a critical review, Int. J. Archit. Comput. 7 (2009) 535–554.

Energy and Buildings 105 (2015) 88–99

Contents lists available at ScienceDirect

Energy and Buildings

journal homepage: www.elsevier.com/locate/enbuild

Building energy optimization in the early design stages: A simplified method

Kristoffer Negendahl*, Toke Rammer Nielsen

Technical University of Denmark, Department of Structural and Civil Engineering, Brovej Building, 118, DK-2800 Kgs. Lyngby, Denmark

ARTICLE INFO

Article history:
Received 5 March 2015
Received in revised form 7 May 2015
Accepted 23 June 2015
Available online 22 July 2015

Keywords:
Energy efficiency
Daylight
Thermal comfort
Building design
BPS
Early design stages
Optimization
Integrated dynamic models
Parametric
Simulation

ABSTRACT

This paper presents the application of multi-objective genetic algorithms for holistic building design that considers multiple criteria; building energy use, capital cost, daylight distribution and thermal indoor environment. The optimization focus is related to building envelope parameters. To obtain relevant feedback from multi-objective optimizations in early design stages, evaluation speed is a key concern. The paper presents a fast evaluation method fit for the early design stages. It uses a combination of two different quasi-steady-state methods for energy and indoor environment evaluations, a Radiance implementation for daylight simulations and a scripted algorithm for capital cost evaluations. The application of the method is developed around an integrated dynamic model which allows visual design feedback from all evaluations to be an integrated part of the design tool experience. It is concluded, that quasi-steady-state methods implemented as part of integrated dynamic models are fast and flexible enough to support building energy-, indoor environment- and cost-optimization the early design stages.

1. Introduction

Building energy optimizations during the early design stages, where information levels are low and design changes are frequent, induce risks of high uncertainty and excessive amount of calculations. Many researchers reason that building performance simulation (BPS) tools in the early design stages is beneficial for building performance such as energy, daylight and thermal indoor environment. However, BPS tools are rarely used in the early design process, consequently optimization with such tools are far from integrated in the early design stage in practice.

Augenbroe [1] argues to better inform the early design BPS tools need to support: (1) A rapid evaluation of designs alternatives, (2) different types of decision making processes and (3) designers' ability to solve nonlinear and multi-criteria problems. Struck [2] supplements that BPS tools must be flexible and fast enough to facilitate changing representations of innovative design concepts thus being able to dynamically scale the model resolution to fit the different information levels. Few tools live up to any such expectations. Simplified BPS tools are fast but only provide simplified feedback while more advanced BPS tools are difficult to use and are often slow in comparison to the simpler tools. Furthermore, only a fraction of these BPS tools can be used in automated processes required to perform building energy optimization. The choice of simplified BPS tools in the early design stages seems to be favored by most practitioners [3]. However, with the purpose of designing with optimization, simplified BPS tools may evidently increase risks of returning inaccurate results, which defies the purpose of using optimization processes in the early design stage. Even though techniques of BPS are undergoing rapid change and dramatic improvements in computing power, algorithms, not feasible only a few years ago [4], the balance between achieving sufficient accuracy and the ability to provide highly flexible and fast feedback to the designer, is still today base for discussion.

In general most methods which apply optimization in early design stages focus on non-geometrical variables such as changing *U*-values, or system requirements and rarely put the analyses in context of project specific architectural solutions. Obviously compulsory and ambitious use of optimization algorithms in the early design stage is of architectural concern. Hermund [5] reacts toward optimization in the design processes:

> Linear working methods that promote the reduction of the creative loops in favor of systemic optimization is one topic that must be addressed by architects . . . Relying on one integrated

* Corresponding author. Tel.: +45 2670 4550.
E-mail address: kristoffer.negendahl@gmail.com (K. Negendahl).

http://dx.doi.org/10.1016/j.enbuild.2015.06.087
0378-7788/

model (referring to IFC- and gbXML-models) could mean an eventual loss of control with real value of the architectural quality: to create meaningful and beautiful spaces for real people. Hermund [5]

The concern of using optimization processes in early design is very real, regardless of how the model is constructed. However, the benefit of optimization may in many cases exceed the downsides of artistic control if the optimization processes is controlled and supervised by the designers themselves. And to counter this problem, geometrical design concepts representing architectural ideas in variations must be easy to integrate with the optimization process. Based on Mora et al. [6] Struck et al. [7] point out such process is supported when the method is able to:

- Assisting rather than automating design.
- Facilitate the quick generation of integrated solutions.
- Shorten synthesis analysis evaluation cycles.
- Support an interaction and selection of most suitable design alternatives.

With the ambition to advance combined qualitative assessments and quantitative optimization in the early design stage, a simplified method to whole building energy optimization is proposed. Based on a real life design problem the article first explains the need for a very fast whole building simulation that could (to an acceptable level of precision) present the whole building energy consumption, the price of the façade, the amount of daylight in every zone and estimate the risk of thermal overheating problems inside the building. All this must be done in a way to make informed feedbacks to the designer on limited amount of information. As a response to these needs this paper shows a new method that allows multi objective optimization with the inclusion of project specific qualitative constraints.

Our approach chooses various simple BPS tools coupled together with a visual scripting tool and results are visualized in the architects design tool. The reasoning to use simple BPS tools over the more complicated and precise simulation tools, are compressed into three requests: (1) to overcome the limited time available in the early design stage, optimization must be as fast as possible. (2) The coupled BPS tools have to fit the early design stage, hence they must be able to make use of the limited amount of information available. And (3) the tools have to fit into an integrated environment that can take the entire design team's expertise into account.

The main focus is on the building envelope optimized for whole building energy consumption, daylight distribution, thermal environment and cost. The method relies on an integrated dynamic model [8] that incorporates a design (CAD) tool Rhinoceros [9] a visual programming language (VPL) Grasshopper [10], the existing BPS tools Radiance [11], Be10 [12] and a new hourly based quasi-steady-state tool (HQSS) to estimate hourly heat gains with the purpose to prevent overheating problems at zone level.

2. Background and related research

Optimization as a process favors limited aspects of a system, which need to be *differentiable in the design parameters* [13] while constraints and objectives need to be clearly defined. Therefore, optimization as a process will often discount those aspects, which has not been included in the cost function. This is arguably the main reason why research in optimization focuses on quantitative performance objectives over qualitative evaluations. Nonetheless many researchers have sought to reconcile the level of artistic control to optimize on predefined criteria with predefined constraints. One example is Petersen [14] who focuses on a list of very specific elements of the particular design instead of aiming for a complete evaluation of every parameter in the early design stage. By limiting the search space the design team saves time in the early design process and optimization may be handled by human thinking alone. However, when design problems grow with design variables and objectives, algorithmic optimization becomes ever more attractive.

To make the design exploration computational feasible to Hopfe and Hensen [15] argued the analysis of sensitive variables is a good starting point for a more integrated design analysis. This of course can be applied to project specific cases that employ stochastic analyses of building models to provide the designer faster indications on which variables are more sensitive or robust. To further speed up this process Hopfe et al. [16] used surrogate modeling techniques to approximate the objective functions on energy consumption and over/under-heating hours. The method used Gaussian processes (Kriging) which correlate quite strongly with the introduced noise on the design parameters, basically to model real-life uncertainties. The idea to use increasingly adaptive surrogate models have also shown promise to include more qualitative assessments (that often means many more design variables) by *listening to design variables and predicting user requests* as suggested by Negendahl et al. [17]. However, this concept has not yet been coupled with optimization algorithms and need further developments in predicting user requests are needed.

Another approach to decrease computationally expensive calculations is to implement adaptive precision control in the BPS tool and approximate cost functions for example Wetter and Polak [18]. This, however require deep access to the solvers precision parameters. In many BPS tools these are fixed at compile time and are hard to access. Nonetheless, Wetter & Polak showed promising results by applying a Hooke-Jeeves optimization algorithm with precision control on a static SPARK model.

Wright et al. [19] showed one of the more recent attempts in applying multi-objective optimization with quality defined constraints into the early design. The design in this context was considered by constraining the geometric proportions of the façade by the golden ratio and visualizing optimal solutions lying on the trade-off between energy use and capital cost. Other efforts to improve the integration of the design process and the energy performance domain include: Caldas [20] and Wang et al. [21] who attempts to involve the more subjective and qualitative objectives into optimization processes. Kim et al. [22] use an agent point strategy to control overall building geometry, this is coupled to a CFD tool and genetic algorithm to optimize wind flow around the building. They considered one building typology and argued that the method would provide design options and *educated intuition for architects to incorporate in design practices*. Gerber and Lin [23,24] showed a prototype tool (*H.D.S Beagle*) to integrate parametric geometry, energy simulation with Green Building Studio and optimization into the early design stage. And finally the ParaGen project [25] by Turrin et al. explored a performance based design process by combining parametric modelling and genetic algorithms correlating structural performance and solar energy. All these methods heavily depend on high computational power and are therefore difficult to use within the limited timeframe of the early design stage.

Ideally faster or even *real-time evaluation speed* like found in the approach of Sanguinetti et al. [26] combined with better quality assurances and implementation of robust optimization methods is to be preferred. Sanguinetti et al. argued for the fast performance feedback as one of the main drivers for designers to explore design alternatives. Their solution was an integration of design synthesis and analysis is implemented through coupling simple parametrically controlled geometric representations generated in a design tool with normative calculations in spreadsheets. The method proved to be highly flexible and could serve project specific design explorations which include almost any qualitative considerations. However they did not show the option to include an optimization algorithm, and did not address the problems of tool validity.

The progress and development of integrated dynamic models [8] where a visual programming languages (VPL) can dynamically couple a design tool to one or more BPS tools have made it easy for non-developers to implement new assessment methods during the early design stages. Integrated dynamic models can assist the building designer in providing performance feedback on sketch like models in the early design stages and automate system designs and other undecided design inputs. Negendahl [8] argues one of the advantages of using integrated dynamic models over e.g. simulation packages is the ability to couple any type and number of BPS tools to the design tool environment. This helps the designer to maintain control of the artistic qualities of the model while receiving visual consequence feedback from the coupled BPS tools within their native design tool. Sargent et al. [27] showed a method to reduce cooling loads by back-tracing rays from different solar angles to construct a 3-dimensional "shading volume" (at room level). The method used an algorithm to calculate the fraction of beam component energy considered desired configuration for the external shading volume. The BPS tool Energy+ was used to evaluate thermal and energy performance. Over existing methods, their method was found more flexible, mainly because of the coupled CAD tool and scripting environment in the integrated dynamic model. With little effort, integrated dynamic models can be coupled with optimization algorithms such is the case of Darwin [28] and Galapagos [29]. These additions to an integrated dynamic model support a wide variety of interaction and selection of most suitable design alternatives. This means integrated dynamic models with optimization algorithms may be one of the better options when seeking to integrate architectural qualities into the optimization process.

The following sections of the article examine how to facilitate quick generations of integrated solutions and shorten the synthesis analyses of evaluation cycles. This especially relates to model speed and type of tools used in the early design stage.

3. Method

3.1. Choice of building performance simulation tools

Table 1 describes three different BPS tools applied in the method; all chosen for their ability to evaluate performance with minimum computational power and dynamically deliver the results back into the model.

Radiance [11] (Evaluation of daylight, Table 1) is processed through the interface Honeybee [30] while Be10 [12] (Evaluation of building energy consumption, Table 1) is processed through the interface Termite [31]. Be10/Termite performs monthly averaged quasi-steady-state calculations and is used in Denmark to evaluate energy consumption of all new buildings. The hourly quasi-steady-state method (shortened HQSS) (Evaluation of thermal overheating, Table 1) is written in python and Grasshopper and is based on ISO 13790 [32].

Table 1
BPS tools applied to the method.

	Objective	Tool	Implementation
1	Evaluation of daylight	Radiance [11]	Ladybug/Honeybee [30]
2	Evaluation of building energy consumption	Be10 [12]	Termite [31]
3	Evaluation of thermal overheating	HQSS	New hourly quasi-steady-state implementation

The monthly calculation performed by Be10/Termite gives accurate results on an annual basis as demonstrated by Christensen et al. [33]. But the results for individual months close to the beginning and the end of the heating and cooling season can have large relative errors [32]. Monthly quasi-steady-state calculations may be sufficient to estimate building energy use but is considered too uncertain as a method to estimate thermal indoor environment. For this reason an alternative quasi-steady-state method for hourly calculations has been added to the model. The HQSS tool facilitates the calculation using hourly user schedules (such as temperature set-points, ventilation modes and hourly control options based on outdoor or indoor climatic conditions). The tool produces hourly results, but similar to other quasi-steady-state hourly calculation methods, the results for individual hours are not validated and individual hourly values can have large relative errors [32]. Nevertheless, for early design stage estimation the use of hourly calculation methods is expected sufficient in detail and precision (more on this statement is discussed in part 7). The HQSS tool is used to estimate an average hourly heat balance to determine whether the cooling load can sustain the internal and external heat gains.

Worth noting is that the computing power of using hourly calculation is around 2 orders of magnitude more intensive than divisional period (e.g. monthly) quasi-steady-state methods. However, this is still at least one order of magnitude less computational intensive than detailed dynamic simulation methods.

3.1.1. Hourly quasi-steady-state method

In the following section the HQSS tool is explained. When considering risk of overheating only few tools presently can evaluate whole buildings fast enough to effectively be used in early stage design processes. The tool is now a part of the Termite plugin for Grasshopper which can be found and downloaded at http://cobalab.dk/

The purpose of HQSS tool is a simple evaluation of cooling capacity efficiency on hourly basis simply by determining the accumulated hours where the cooling capacity $Q_{C,cap}$ does not meet the heat loads Q_{load} at each calculation step t:

$$\sum_{t=1}^{nt} \left(Q_{C,cap} \leq Q_{load}\right) \tag{1}$$

where $Q_{C,cap}$ is the cooling capacity and Q_{load} is the heat loads at any calculation step t, t is defined as 1 h in the range of a year of 8760 h. However, to speed up the calculation process the amount of calculation steps, nt is reduced in two ways. (A) Only hours, t within the service period (usage profile) of the given zone are considered, in this case as an office open [08–17] every day, all year. (B) Only hours, t where direct solar irradiance has an effect on the given zone are considered, see Eq. (8).

Each building zone for each calculation step the total heat transfer, Q_{ht} is given by [32]

$$Q_{ht} = Q_{tr} + Q_{ve} \tag{2}$$

where Q_{tr} is the total heat transfer by transmission and Q_{ve} is the total heat transfer by ventilation.

The total heat gains are expressed as

$$Q_{gn} = Q_{int} + Q_{sol} \tag{3}$$

where Q_{gn} is the total heat gains for each calculation step, Q_{int} is the sum of internal heat gains, and Q_{sol} is the sum of solar heat gains over the given period.

The ideal cooling demand in any point in time where the sum of heat gains are larger than the sum of (positive) heat transfers can be expressed as

$$Q_{C,\mathrm{nd,cont}} = Q_{\mathrm{gn}} - \eta_{C,\mathrm{ls}} \times Q_{\mathrm{ht}} \tag{4}$$

where $Q_{C,\mathrm{nd,cont}}$ is the needed amount of cooling to maintain set point temperatures and $\eta_{C,\mathrm{ls}}$ a dimensionless utilization factor depended on time constants and used specifically in seasonal and monthly calculation periods [32]. When the maximum cooling capacity, $Q_{C,\mathrm{cap}}$ is known, Eq. (4) can be written as

$$Q_{C,\mathrm{cap}} \geq Q_{\mathrm{int}} + Q_{\mathrm{sol}} - (Q_{\mathrm{tr}} + Q_{\mathrm{ve}}) \tag{5}$$

The internal gains, Q_{int} for each zone k in each calculation step t can be extracted as

$$Q_{\mathrm{int}} = \left(\sum_{k=1}^{nk} \left(Q_{\mathrm{equip},k} + Q_{\mathrm{occu},k} + Q_{\mathrm{light},k} \right) \right) t \tag{6}$$

where k is the zone and nk is the number of zones in the building, $Q_{\mathrm{equip},k}$ = 6 W/m^2 and $Q_{\mathrm{occu},k}$ = 4 W/m^2 are assumed constant in every calculation step t (since only the service period is considered). $Q_{\mathrm{light},k}$ is calculated as the interpolated value based on a daylight factor, DF from radiance (see Eq. (13).) The daylight factor is reduced to; if DF > 3%=3% and the effect $Q_{\mathrm{light},k}$ is normalized to fit the range [0.3] % with the expression:

$$Q_{\mathrm{light},k} = \left(\frac{(\mathrm{DF} - 0\%) \times \left(Q_{\mathrm{light,max},k} - Q_{\mathrm{light,min},k} \right)}{3 - 0\%} \right) + Q_{\mathrm{light,min},k} \tag{7}$$

The solar gains, Q_{sol} is assumed to be composed of a direct beam component depended on solar position v, and a constant diffuse component depend on the sun position in the calculation step t;

$$Q_{\mathrm{sol}} = \left(\sum_{w=1}^{nw} \left(\sum_{v=1}^{nv} (\cos(\varphi_v) \times I_v + Q_{\mathrm{dif}}) \times g \times b \times A_w \times FR \right) \right) t \tag{8}$$

where w is the window in a façade and nw is the number of windows in the zone, v is the unique sun vector visible from the window and nv is the total amount of vectors. φ is the incidence angle to the sun vector, and I_v is the correspondent (beam component) effect from the sun. g is the g-value of window pane, b is an adjustment factor which is further described in the discussion, A_w is the window area and FR is the frame ratio, Q_{dif} is the diffuse contribution calculated to: 70 W for the particular site. Q_{dif} is estimated as an average fraction of horizontal diffuse radiation, Dh with the function;

$$Q_{\mathrm{dif}} = \bar{Dh} \times \left(180 - \frac{\beta}{180} \right) \tag{9}$$

where β is the inclination angle of 90°.

The solar gains evaluation is defined as annual simplified solar beam component simulation. To speed up the calculation process the annual hourly sun vectors are reduced from 8760 to 103 vectors, while the irradiance effect, I_v per unique sun vector, v is maintained in every vector group. As a consequence, each original placed vector is repositioned slightly on the hemisphere (see Fig. 1). While this will affect the angle of incidence, φ_v, the precision of the calculations are only slightly biased in the process, more on this subject is found in the discussion. If unobstructed each irradiance factor with the new angle of incidence for each window are calculated. However most sun vectors are obstructed by the building geometry which means most sun vectors are omitted from the calculation,

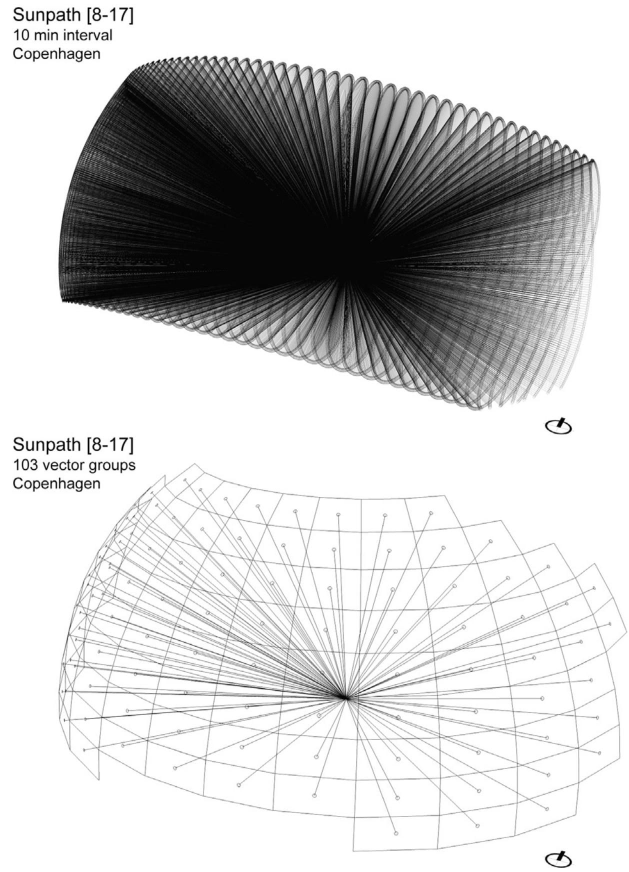


Fig. 1. Annual solar sky component generated from the usage profile of a typical office [8–17]. The reduced vector field can be seen in the bottom picture.

this again makes calculations run significantly faster. The obstruction calculation is processed by an isovist[1] [34] function.

The transmission losses/gains, Q_{tr} for each zone k in the each calculation step t are extracted as

$$Q_{\mathrm{tr}} = \left(\sum_{k=1}^{nk} A_{\mathrm{win},k} \times U_{\mathrm{win},k} + A_{\mathrm{wall},k} \times U_{\mathrm{wall},k} + l_{\frac{\mathrm{win}}{\mathrm{wall}},k} \times \psi_{\frac{\mathrm{win}}{\mathrm{wall}},k} \right) \times \left(\theta_{\mathrm{set},i} - \theta_e \right) t \tag{10}$$

where $A_{\mathrm{win},k}$ is the area and $U_{\mathrm{win},k}$ is the U-value of the window (inclusive frame), $A_{\mathrm{wall},k}$ and $U_{\mathrm{wall},k}$ is the area and U-value of the opaque part of the façade. $l_{\mathrm{win/wall},k}$ and $\psi_{\mathrm{win/wall},k}$ is the length and transmission factor of the connection between wall and window. The cooling set point temperature $\theta_{\mathrm{set},i}$ is assumed 26 °C, infiltration is ignored and θ_e is the external temperature at calculation step t.

The ventilation loss/gains, Q_{ve} for each zone k in the each calculation step t are extracted as

$$Q_{\mathrm{ve}} = \rho_{\mathrm{air}} \times c_{\mathrm{air}} \times \left(\sum_{k=1}^{nk} b_{\mathrm{ve},k} \times q_{\mathrm{ve},k,\mathrm{max}} \right) \times \left(\theta_{ve,\mathrm{set},i} - \theta_e \right) t \tag{11}$$

where $\rho_{\mathrm{air}} \times c_{\mathrm{air}}$ is the heat capacity of air volume set to 1200 J/(m^3 K), $b_{\mathrm{ve},k}$ is the dimensionless temperature adjustment factor representing the heat recovery rate. $q_{\mathrm{ve},k,\mathrm{max}}$ is the maximum airflow expressed in m^3/s. The air supply temperature $\theta_{\mathrm{ve,set},i}$ is assumed to be 18 °C and θ_e is the external temperature at calculation step t. Please notice that the part of the cooling capacity

[1] Isovist is defined as an object that can be seen from a given point in space.

related to cooling outside air to 18 °C is not accounted for in the minimization function seen in Eq. (15).

3.2. Choice of optimization method

During the past decade, design optimization using performance simulation has been associated with stochastic methods such as Simulated Annealing e.g. [35] and Genetic Algorithms e.g. [36] and Gradient-based methods e.g. [37]. Many methods applies to design problems for optimizing thermal and lighting performance, based on building enclosure, HVAC design, and control schedules, as mentioned in [18,38,39]. As Wetter [13] explains there are several challenges in using BPS tools in combination with stochastic optimization algorithms. Stochastic optimization algorithms are computationally efficient (over their deterministic counterpart), but they often require the cost function to be differentiable in the design parameters. And since many BPS solvers approximate solutions due to adaptive variations in solver iterations [18], the solvers form discontinuous search spaces, which are often difficult for stochastic optimization algorithms to handle. Beside the careful choice of an optimization algorithm, the way the optimization algorithms maintain support of feedback process among different professions in the design team during design iterations are of great importance [7]. To support the early design stage, the method need to facilitate quick generation of integrated solutions and shorten synthesis analysis evaluation cycles as described by Struck [2]. In the same time, the method should allow interaction with the most suitable design alternatives, as well as assists rather than automate design.

As the method relies on an integrated dynamic model [8], it enables exploration of different design options by adding visual scripting options. When used in combination with a multi-objective optimization algorithm, multiple designs can be generated and evaluated automatically within the set parameter constraints, with high scoring designs identified and stored [40].

Many (multi objective) methods e.g. [23,36,39,41–43] utilize a variation of a Pareto[2] Ranking of the objectives. Often this does not in self ensure interaction with the most suitable design alternatives, however the ranking method allow an easy way to identify a set of feasible designs that are equal-rank optimal. Arguably optimization of multivariate problems like building design, competing criteria are un-evenly balanced, and their relative importance is generally not definable. Therefore, the use of non-dominated ranking methods, help the design team to navigate in the infinite space of solutions [44].

A recent implementation, Octopus [45] of the SPEA2 [46] algorithm is both user friendly and flexible enough to integrate into most design optimization processes. Octopus/SPEA2 has been used in the application seen in section 5.

4. Case study—Application of the method

4.1. Problem definition and constraint functions

This case study is based on an undisclosed project between the architects BIG and the consultant agency Grontmij. The case is used to present the application of the method in real life design problems where architectural qualities may supersede other objectives.

From the very beginning the design team sought to avoid external solar shadings as solar shadings were found to be expensive, difficult to maintain and difficult to incorporate in the architectural design. The design team argued that most, if not all, external solar shading systems could be avoided by carefully designing a self-shading (folded) façade (see Fig. 2).

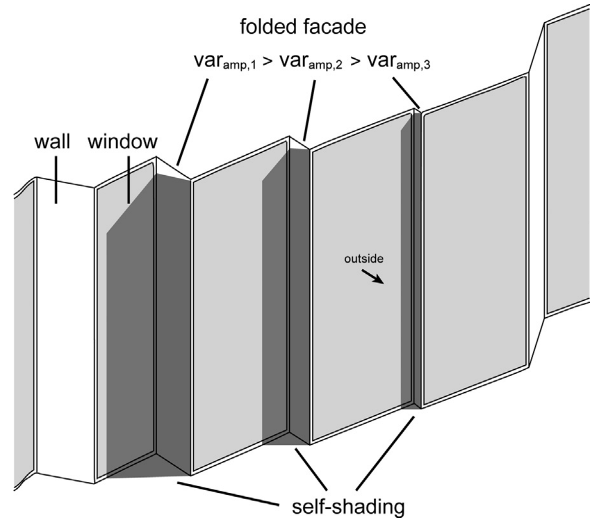


Fig. 2. Folded facade concept. The amplitude of the facade folds marked by the variable $var_{amp,1-3}$ create self-shading mechanisms on the neighboring facade unit.

By removing the external shading system as a viable design option, concerns of thermal indoor environment, building energy consumption and daylight distributions became a central part of the discussion. Four questions arose with the folded façade concept:

(1) How much folding[3] is needed to avoid overheating?
(2) Does increasing amplitude of folds, var_{amp} (see Fig. 2) decrease the energy consumption?
(3) If so does it pay off to use more expensive high performing glazing types[4]?
(4) How does the folding affect the daylight distributions in the offices?

To answer these questions, it was decided to make use of a multivariate optimization method to explore the many solutions where folding could influence the energy consumption, the daylight distributions and indoor thermal environment while considering the cost of the window systems.

It was decided to use a whole building evaluation process of energy, cost, daylight and thermal indoor environment as the architects wanted to control a continuous and changing façade around the building. Using simple representatives of rooms (e.g. by simulating variations of rooms) was found to be unfitted for this process as the small and continually connected variations on the façade would create too many possible combinations and thus too many simulations. What was needed was a very fast whole building simulation that could (to an acceptable level of precision) present the whole building energy consumption, the price of the façade, the amount of daylight in every room and estimate the risk of thermal overheating problems inside the building. To do this, the building needed to be divided into thermal zones and simulating each zone would be necessary, however at this point in the design process room placements were not fixed which meant any zone division

[2] Vilfredo Pareto (1848–1923) developed the concept known as 'Pareto optimality', which is defined by its "equilibrium of positions, from which it is not possible to move so as to increase the utility of some entity without decreasing the utility of another entity." [44].

[3] Amount of folding is determined by adjusting amplitude var_{amp} and window size var_{pl} (see Fig. 3) and var_{blend} (see Fig. 4).

[4] High performing glazing types: window panes with reduced convection and radiation heat losses (low U-values) and/or reduced solar heat gain coefficients (low g-values).

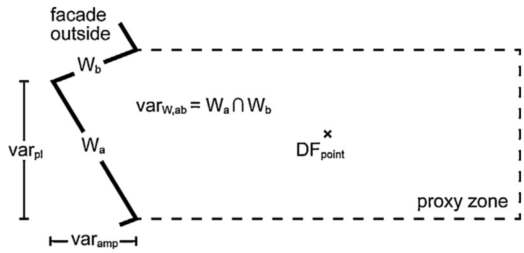


Fig. 3. Plan view of a proxy zone represented as a dashed line. The variables are used to constrain the optimization process.

were very dubious and would affect the simulations significantly. It was for this reason decided to use proxy zones instead of actual room geometry. The proxy zone as seen in Fig. 3 is defined by a volume extruded into the building in a fixed depth (here 5 m) from the façade where the window W_a or W_b is positioned.

As mentioned before, the architects valued a continuous façade, where one fold were mostly similar to the neighboring folds, which meant only subtle changes from façade fold to the next was allowed. In terms of optimization, this is a complex type of dynamic constraint. However, the implementation of this type of constraint functions is straight forward when VPL's are present in the model environment. The design team's solution is a scripted function that utilizes the hyper parameters $\text{var}_{\text{amp}(1-3)}$, $\text{var}_{\text{pl}(1-3)}$, $\text{var}_{\text{blend}}$ to control the folding. Where $\text{var}_{\text{amp}(1-3)}$ controls the amplitude in on the three facades. $\text{var}_{\text{pl}(1-3)}$ controls the vertical placement of the fold on each façade and $\text{var}_{\text{blend}}$ adjust the "blending effect", that intermix the folding between facades. Fig. 4 shows variations of the hyper parameters for example var_{pl} shifts the fold clockwise with small values and $\text{var}_{\text{amp}(3)}$ controls the north eastern façade.

By defining these geometrical constraints, the idea was to explore the many different "optimal" solutions that were provided from the optimization process. The different solution showed in Fig. 4 does not represent any architectural preferred strategy, but shows the impact of the design variables.

4.2. Objective functions

Four objective functions $f_{\text{energy}}(x), f_{\text{cost}}(x), f_{\text{daylight}}(x), f_{\text{thermal}}(x)$ are minimized by the multivariate optimization algorithm SPEA2 [45,46].

The building energy use; $f_{\text{energy}}(x)$ is a function of the annual simulated heating Q_{heating}, cooling Q_{cooling}, ventilation Q_{vent} and lighting Q_{light}:

$$\min_{x\in\mathbb{R}^s\times\mathbb{Z}^t} f_{\text{energy}}(x) = \sum_{i=1}^{ni}\left(Q_{\text{heating},i} + \text{PF}\times\left(Q_{\text{cooling},i} + Q_{\text{vent},i} + Q_{\text{light},i}\right)\right) \quad (12)$$

where i is the load condition of the particular condition, ni and is the number of load conditions. $f_{\text{energy}}(x)$ is simulated by Be10 [12] through the Termite [31] interface. The primary energy factor, PF = 2.5 is multiplied with electrical energy uses according to the Danish building regulations [47].

The capital cost of the façade is a function of the cost of the transparent parts of the façade: Cost index shown in Table 2 is generated for this article and should not be used in general. Seven different window types were considered each evaluated by their cost index and amount of glazed areas in the particular solution, x:

$$\min_{x\in\mathbb{R}^s\times\mathbb{Z}^t} f_{\text{cost}}(x) = \left(\sum_{k=1}^{nk}\frac{(A_{w,k}\times C)}{10}\right) + 50 \quad (13)$$

where k is the proxy zone and nk is the number of proxy zones in the building, $A_{w,k}$ is the window area in the k'th proxy zone and C is the cost index see Table 2. The constants 10 and 50 are unitless and added to normalize the relative objectives seen in Figs. 6 and 7.

The daylight evaluation $f_{\text{daylight}}(x)$ is defined by the CIE uniform sky simulation of a point in the center of the proxy zone, 0.85 m from the floor. A penalty function $\text{ERF}(\text{DF}_w) - 3$ (also shown in Fig. 5) based on the Gauss error function, ERF [48] is used to reduce the importance of very high daylight factors and increase the penalty of DF < 3% (the penalty function related to daylight factors can be seen in Fig. 5):

$$\min_{x\in\mathbb{R}^s\times\mathbb{Z}^t} f_{\text{daylight}}(x) = \frac{nk}{\left(\sum_{k=1}^{nk}\left(\text{ERF}\left(\text{DF}_{w,k}\right) - 3\right)\right)} \quad (14)$$

where k is the proxy zone and nk is the number of proxy zones in the building. $\text{ERF}(\text{DF}_w)$ is defined by $\text{ERF}(\text{DF}_w) = \frac{2}{\sqrt{\pi}}\int_0^{\text{DF}_x} e^{-t^2}\,\text{d}t$. DF_x is simulated by Radiance for every solution, x.

The objective function of the thermal requirements is defined as follows.

In each calculation step t evaluate:

if $\left(Q_{\text{int}} + Q_{\text{sol}} - \eta\times(Q_{\text{tr}} + Q_{\text{ve}}) \geq \theta_{C,\text{cap}}\right)$ is True increment overheating hour h

$$\min_{x\in\mathbb{R}^s\times\mathbb{Z}^t} f_{\text{thermal}}(x) = \frac{nk}{\left(\sum_{k=1}^{nk}\sum_{t=1}^{nt}(h)\right)} \quad (15)$$

where k is the proxy zone and nk is the number of proxy zones in the building, $Q_{\text{int}} + Q_{\text{sol}} - \eta\times(Q_{\text{tr}} + Q_{\text{ve}}) \geq Q_{C,\text{cap}}$ is explained in Eq. (5). $\theta_{C,\text{cap}} = Q_{C,\text{cap}}/A_{\text{proxy}}$ represents the maximum cooling capacity at any hour in the year, set to 40 W/m^2. A_{proxy} is the area of the proxy zone.

5. Results

The multivariate optimization procedure was performed at dual-core laptop over a period of 3 days. A population size of 300 ran through 32 generations of SPEA2 [46] trials, which turned out to be sufficient for convergence. In average every simulation/evaluation of the four criteria took less than 30 s. This is considered very fast when we are talking whole building simulations on regular PCs.

In Fig. 6 all the most promising solutions are showed. The green colored boxes represents the solutions with minimum amount of thermal loads (hours above the maximum cooling capacity see Eq. (15)) in the 32nd generation of simulations. The red colored boxes are the worst performing solutions in terms of thermal loads. The grey boxes are the Pareto solutions in generation 1–31. From the figure it can be seen that several cluster developments occur in boomerang-like fields around the shared minimum (0, 0, 0, 0).

Each field seen in Fig. 6 is a separate solution space for the window types (seen in Table 2). It can be concluded that anyone of the seven window types can be used in the building, however type 2 is in general least costly (in terms of capital costs) and type 5 is the most expensive of the seven window types. This is interesting as the cost-distribution do not follow the cost-index shown in Table 2.

When looking more specifically into the Pareto solutions (see Fig. 7) of the final generation a wide range of folded façade compositions can be seen. From the figure it can be seen that every one of the selected solutions, except of solution 3, has windows on the right side (seen clockwise from the top) of the folded façade. All solutions, but solution 1 tend to open up with more glazing toward north east and close itself toward south east.

Only solution 5 seems to have a complete uniform façade around the building, all the other solutions have individual façade compositions for the three orientations. Solution 3 ranks highest in terms of daylight (1.1), but worst in terms of thermal loads (234.3).

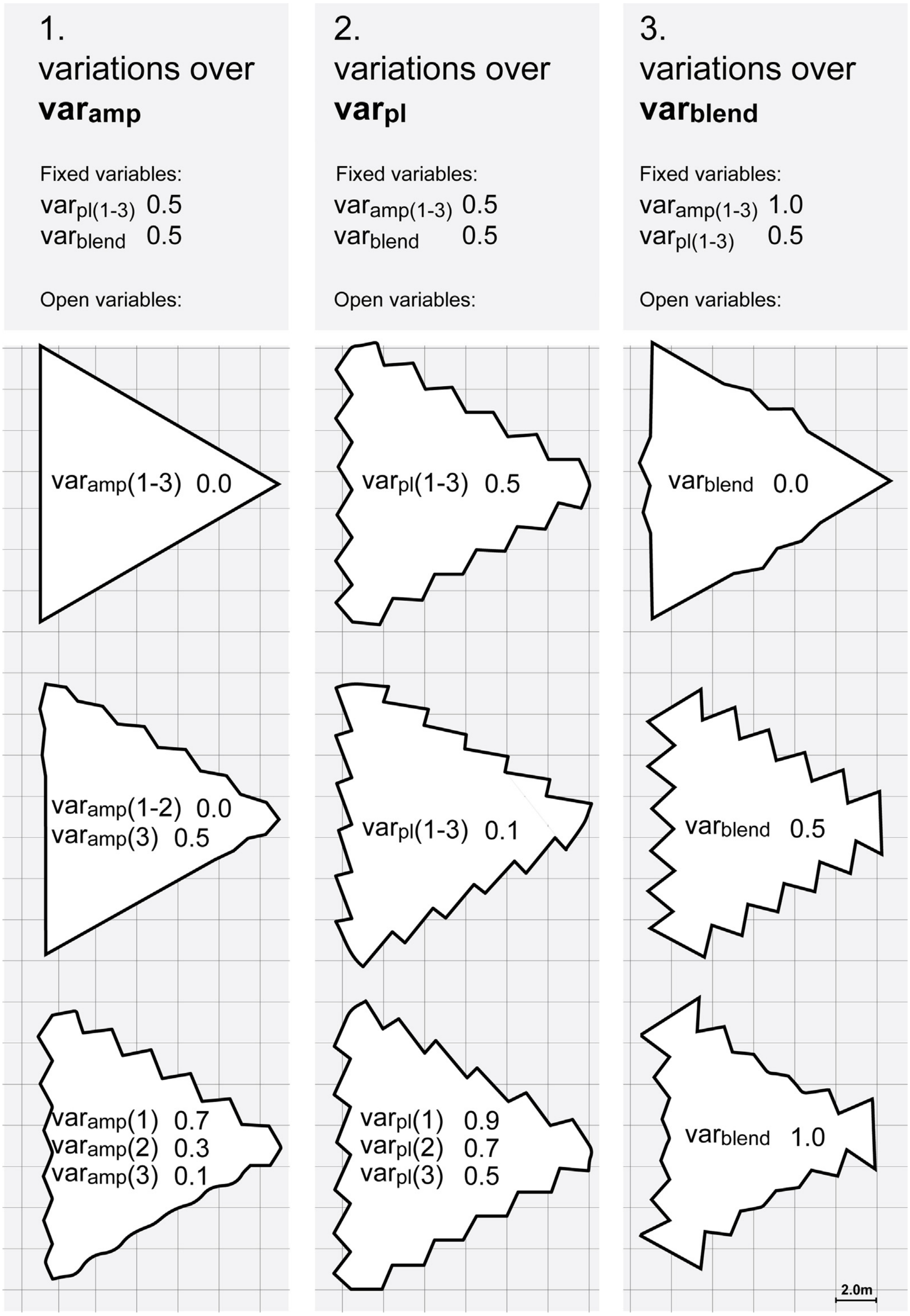


Fig. 4. Plan views of a small building example to explain the changes in design variables. (1) Shows variations over var_{amp}. (2) Shows variations over var_{pl}. (3) Shows variations over var_{blend}.

This also correlates to the usual assumptions of the reversed performance relationship between daylight conditions and a stable thermal environment. Solution 1 performs best in terms of energy performance (154.3) while solution 6 is performing worst in terms of energy (164.9).

When it comes to cost-benefit analysis of the seven selected solutions the capital cost versus running costs (building energy consumption) is a popular way to choose a particular balanced solution. From Fig. 8 the tradeoff between cost (of windows) and cost of annual building energy use is seen. The seven solutions are spread out in the solution space, however solution 5 is performing significantly better in terms of the cost-tradeoff than the others.

It is up to the design team to choose which overall tradeoff-strategy that suits the design better. The seven choices of solutions shows that a very diverse façade composition with a large amount of folds may be optimal if daylight and thermal environment is

Table 2
Window type properties, *Cost index is created for this case study and do not signify real costs.

Name	Configuration	U_{win} (EN10077-1)	U_g (EN673)	G (EN410)	LT (EN410)	E_{ref} (DS418)	Cost index*, C
GlzType1	4-16-4 Argon	1.33	1.02	0.52	0.77	−34	1
GlzType2	4-16-4 Argon	1.40	1.11	0.62	0.80	−24	1.1
GlzType3	4-16-4 Argon	1.53	1.26	0.77	0.82	−11	2.6
GlzType4	4-12-4-12-4 Argon	1.11	0.72	0.50	0.72	−17	1.9
GlzType5	4-12-4-12-4 Argon	0.79	0.5	0.42	0.63	−2	3.6
GlzType6	4-12-4-12-4 Argon	0.82	0.53	0.50	0.72	8	4.7
GlzType7	4-12-4-12-4 Argon	0.90	0.62	0.62	0.73	21	6.1

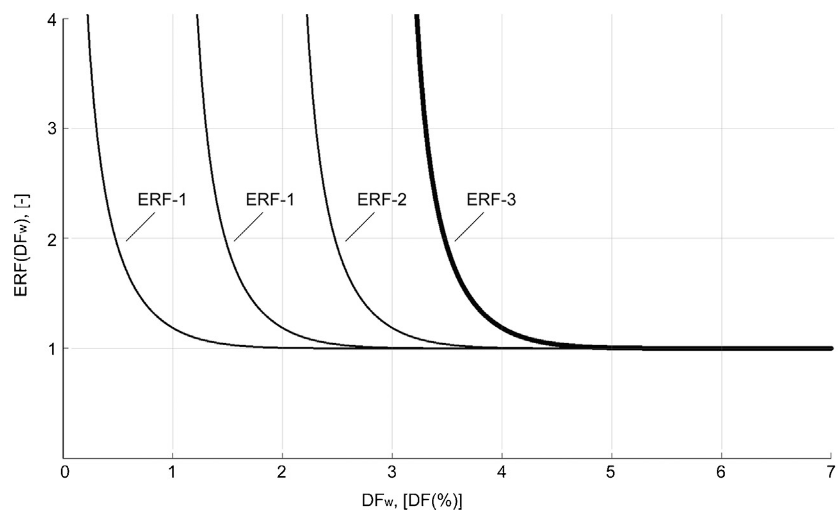


Fig. 5. Penalty functions used to limit the influence of very high daylight factors and avoid low daylight factors. Penalty factor, ERF-3 is used in the case study.

valued high, but in terms of capital-cost and annual energy costs a uniform and almost flat façade composition is better performing.

6. Discussion and future research

This article considers a wide range of problems when BPS tools are used to optimize buildings in the early design stages. One is the actual use of optimization methods in early design stages, which clearly has its limitations, as machine automation is very difficult to combine with quality-defined objectives. Souza et al. [49] warned that the distance between those that simulate and those that design may be one of the largest problems when using optimization methods in early design stages: Setting up criteria to evaluate performance and relate these criteria directly to design actions is a methodological problem independent of the simulation tool being used. It requires simulationists to fully understand the way designers think, i.e. essentially exploring interactions of all parameters together and dealing with all the variables at the same time [49].

To a great extent, this can be solved by utilizing an integrated dynamic model where both the simulationists and the designers work in a fully coupled environment [50]. In our case, the design team that consists of designers and simulationists were able to develop an integrated dynamic model that took both qualitative and performance based criteria into account. This evidently leverages some of the quality assurances mentioned by Hensen [4], such as using appropriate levels of model resolution for the early design stage and requirement for sufficient domain knowledge by the users. However, in terms of the use HQSS to estimate thermal loads, it was performed through a non-validated software tool. Therefore we will provide further details of the method here in the discussion.

The model itself were part of the design process that contributed in the decision making of how to design the façade, therefore we do not consider the optimization method as a definite form finding process but more as mean to extract valuable information from an open ended design problem. The facilitation of performance feedbacks of individual design solutions between the parties in the design team was at no point an issue since the model were operated by both the simulationists and the designers. In relation to facilitation speed and the method's ability to *shorten synthesis analysis evaluation cycles*, as noted by Mora et al. [6] and Struck et al. [7], the integrated dynamic model was able to generate a new result in less than 30 s on a fairly modest two-core laptop. The flexibility of the integrated dynamic model meant that the objectives and constraints of the optimization could be adjusted to fit the design process and not the other way around. Even though much of process of generating solutions was part of automation processes, the actual value of the method is found in the consequence feedback. Or put in another way the value is found in the facilitation of the design rather than in the automation of the design.

The BPS tools used by the model are integrated and fast, but it comes with a cost of validity and precision. The annual energy simulations based on Be10 are merely presenting a trend in energy consumption when the geometry in the model is changed in marginal steps. Of this reason small façade changes will not affect the energy use significantly. The dynamic effects of building use e.g. pulling down curtains when the sun creates glaring effects in offices, is not taken into account. And many similar dynamic effects, which are not considered, may result in inaccurate daylight and energy evaluations.

The thermal indoor environment is estimated from hourly heat balance equations, which ignores thermal accumulation. This assumption is the single most significant source of errors in the model. To counter this in future implementations, thermal capacities and dynamic effects need to be considered. Furthermore HQSS assume constant internal loads (apart from light $Q_{light,k}$). In reality internal loads these will vary much during the service hours, particularly the occupancy. Therefore, we see further improvements in

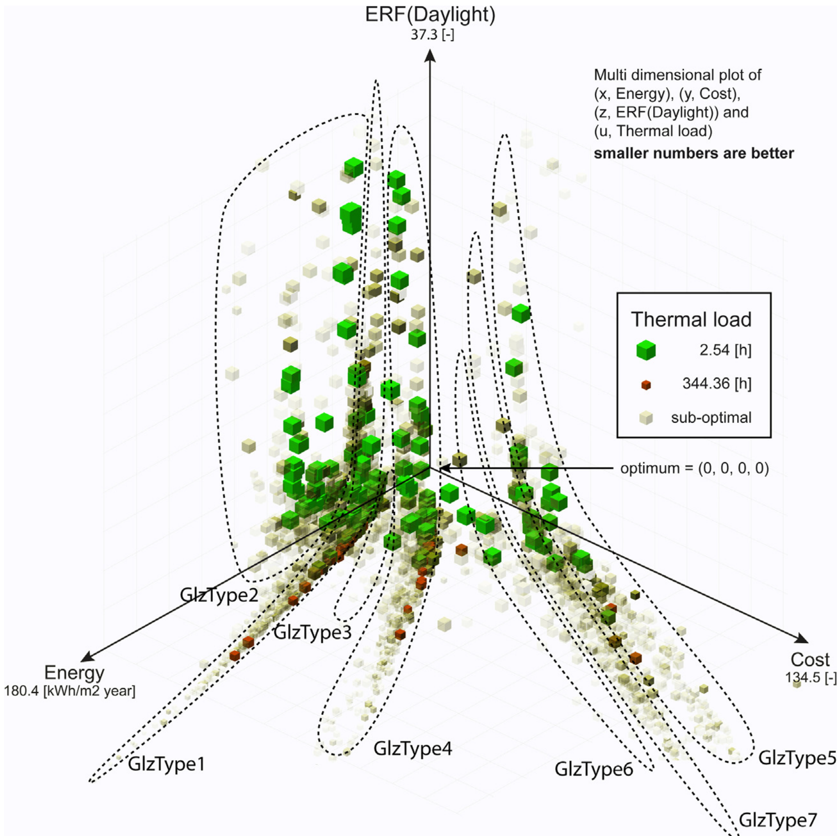
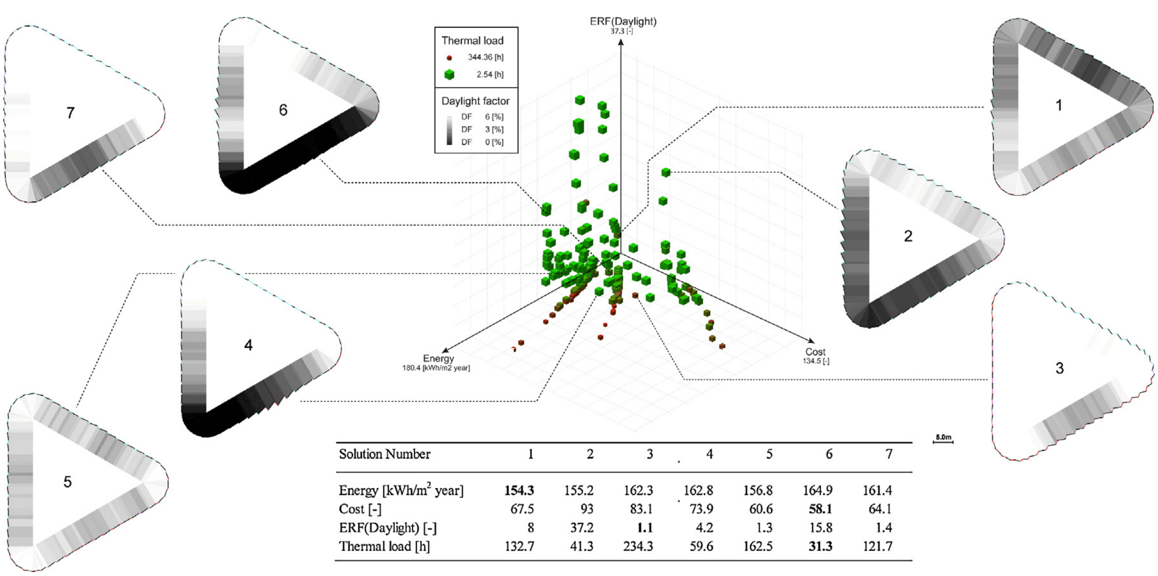


Fig. 6. Four dimensional solution space: Energy, Cost, ERF(Daylight) and Thermal load. Sub-optimal solutions are shown in gray colors. Dashed lines encapsulate the solutions associated with individual window-types. Please see Fig. 5 for the explanation of ERF and Fig. 7 for the explanation of size and color of solutions. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



Solution Number	1	2	3	4	5	6	7
Energy [kWh/m^2 year]	**154.3**	155.2	162.3	162.8	156.8	164.9	161.4
Cost [-]	67.5	93	83.1	73.9	60.6	**58.1**	64.1
ERF(Daylight) [-]	8	37.2	**1.1**	4.2	1.3	15.8	1.4
Thermal load [h]	132.7	41.3	234.3	59.6	162.5	**31.3**	121.7

Fig. 7. The plot in the middle shows axes of a Cartesian space (x, y, z), where x is energy [kW h/m^2 year], y is Cost [–] based on the cost function of windows and z is ERF(Daylight), which represents the penalized function of Daylight factors ERF(DF%). The box size and color describe the amount of hours [h] above the maximum cooling capacity. The plan view of 7 selected solutions are shown in the solution space, daylight factors in each zone are plotted as a grey scale hatch. The table in bottom shows details on the objectives for the selected solutions.

load profiling and incorporation of dynamic occupancy loads. However, these improvements must be implemented in a way that has little effect on the calculation intensity to maintain short evaluation cycles.

The reduced number of calculations per zone is primarily due to the reduced number of solar vector calculations as shown in Fig. 1.

The consequence of altering the solar vector angles is shown in Fig. 9 where the number of “critical” sky subdivisions is

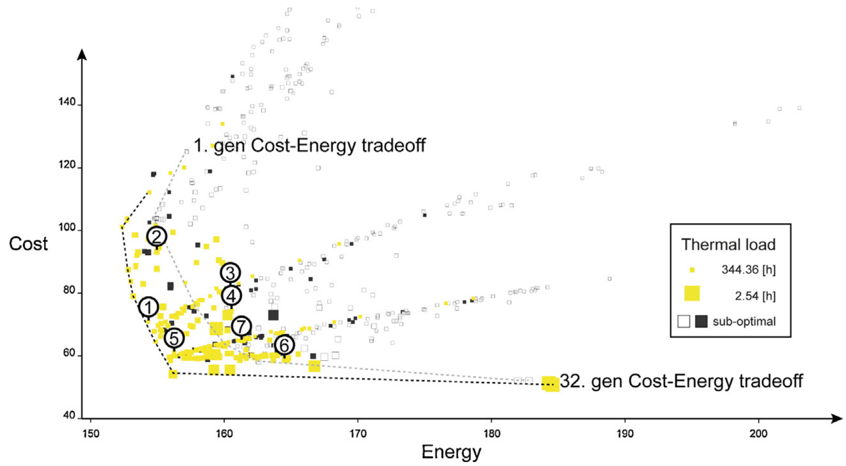


Fig. 8. Tradeoff between energy and cost plotted with sized points representing thermal loads. 1st generation Pareto front of solutions are shown with a gray line and the 32nd generation Pareto front are shown with a black line.

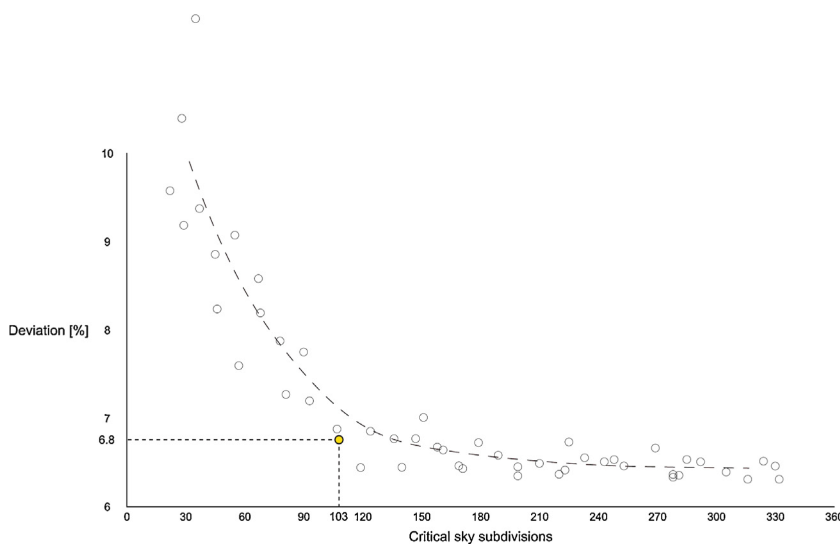


Fig. 9. Absolute beam component deviations between Energy+ and HQSS in % when altering the critical sky subdivisions. Sky subdivision used in this article is marked in the plot.

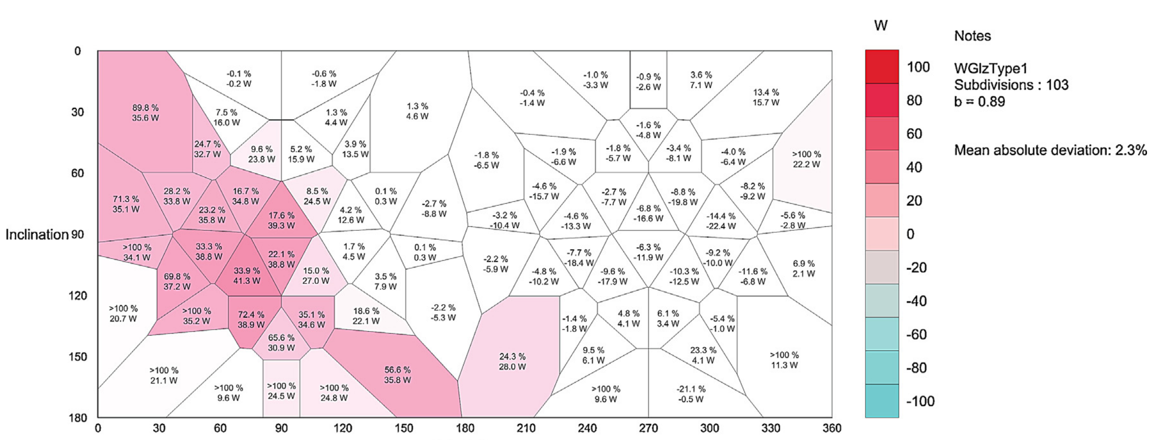


Fig. 10. Beam component deviations between Energy+ and QHSS measured in Watts plotted against inclination and orientation of a surface.

compared to Energy+. As seen from the figure some of the subdivisions are more likely to be similar to the Energy+-results and 103 subdivisions induce a fairly modest deviation of 6.8% compared to Energy+. To further reduce deviation from Energy+ an adjustment factor, b is implemented. The factor is numerically fitted to several Energy+ simulations (with varying window properties seen in Table 2). The comparison of simulations were performed on a sphere with a high angle division which means that comparisons is considered from beam component contribution from the entire hemisphere (one example is shown in Fig. 10). The particular site, weather data, window types and usage profile have resulted in an average adjustment factor, b of 0.89. It can be seen from Fig. 10 that HQSS deviations from Energy+ are varying over the orientation and inclination with a bias toward east around the vertical inclination angle. The absolute mean deviation between HQSS and Energy+ is 2.3% when adjustment factor, b of 0.89 is included.

The diffuse sky contribution has been calculated by assuming average isotropic radiation from the whole sky dome, as it follows (from Eq. (9)). Thus assuming that Q_{dif} only relies on the window tilt of the angle β that receives a proportional part of D_h. However, diffuse radiation is not uniformly spread across the sky. For instance, the area just around the sun (*circumsolar*) is considerably brighter than the rest of the sky. A commonly used method to model this is the Perez model [51]. To further improve the precision of the HQSS future implementation should consider the dynamics of diffuse lighting component. HQSS or similar quasi-steady-state methods should be used with care if actual overheating hours, as demonstrated here, is needed for authenticating purposes. However, for early design stage indications, these tools are found to be sufficient in terms of detail and precision. Nevertheless, more research on this topic is necessary.

7. Conclusions

As demonstrated, multivariate optimization combined with simplified building performance tools leads to the finding of optimal solutions in reasonable computational time. It is clear that an integration of optimization algorithms can drastically change the usage of time within architectural design processes, allowing designers to focus their attention on taking informed design decisions. It is concluded, that quasi-steady-state methods implemented as part of integrated dynamic models are fast and flexible enough to support building energy-, indoor environment- and cost-optimization the early design stages. Additionally these types of models showed potential to integrate various types of architectural constraints in the optimization process, thereby integrating the domains of the building designer and the simulationist through a common platform.

For the particular application of the method, it is concluded that a wide variety of solutions may be feasible. In terms of how much a façade should fold, the choice of window type and window size is the determining factors.

As a final note on validity and precision on the demonstrated method, the use of an hourly quasi-steady-state method for estimating thermal problems should only be used in determining the direction of design, rather than the final design. The same is concluded with the use of a monthly quasi-steady-state method for estimating whole building energy use. The estimations of daylight conditions and capital cost of the façade is found valid even in later design stages. When it comes to using the combined evaluations with stochastic optimization algorithms (like the SPEA2 algorithm demonstrated), it can be concluded the level of precision is sufficient for the initial design approach, but more precise evaluation methods are needed in later stages when more detailed design options has been settled.

Acknowledgments

The work is part of PhD research currently held at the Technical University of Denmark, and is performed in collaboration with Grontmij and Innovationsfonden. A special acknowledgement goes to BIG.

References

[1] G. Augenbroe, Trends in building simulation, Build. Environ. 37 (2002) 891–902.
[2] C. Struck, Uncertainty Propagation and Sensitivity Analysis Techniques in Building Performance Simulation to Support Conceptual Building and System Design, Eindhoven University of Technology, 2012, http://repository.tue.nl/735575, ISBN 978-90-386-3242-1.
[3] S. Attia, J.L.M. Hensen, L. Beltrán, A. De Herde, Selection criteria for building performance simulation tools: contrasting architects' and engineers' needs, J. Build. Perform. Simul. 5 (2012) 155–169.
[4] J.L.M. Hensen, Towards more effective use of building performance simulation in design, in: Proc. Seventh Int. Conf. Des. Decis. Support Syst. Arch. Urban Planin., 2004, pp. 2–5.
[5] A. Hermund, Building information modeling in the architectural design phases: and why compulsory BIM can provoke distress among architects, in: G. Cagdas, B. Colakoglu (Eds.), COMPUTATION: The New Realm of Architectural Design: Proceedings of the 27th Conference on Education and Research in Computer Aided Architectural Design in Europe, vol. 27, 2009, pp. 75–81, COMPUTATION: the new realm of architectural design, no. 27.
[6] R. Mora, C. Bédard, H. Rivard, A geometric modelling framework for conceptual structural design from early digital architectural models, Adv. Eng. Inf. 22 (2008) 254–270.
[7] C. Struck, P.J.C.J. de Wilde, C.J. Hopfe, J.L.M. Hensen, An investigation of the option space in conceptual building design for advanced building simulation, Adv. Eng. Inf. 23 (2009) 386–395.
[8] K. Negendahl, Building performance simulation in the early design stage: an introduction to integrated dynamic models, Autom. Constr. 54 (2015) 39–53.
[9] Robert McNeel & Associates, Rhino, 2013, ⟨http://www.rhino3d.com⟩.
[10] Robert McNeel & Associates, Grasshopper3d, 2013, ⟨http://www.grasshopper3d.com⟩.
[11] G.J. Ward, F.M. Rubinstein, R.D. Clear, A ray tracing solution for diffuse interreflection, ACM SIGGRAPH Comput. Graph. 22 (1988) 85–92.
[12] SBI, Be10, 2013, ⟨http://www.be10.sbi.dk/⟩.
[13] M. Wetter, A view on future building system modeling and simulation, in: J.L.M. Hensen, R. Lamberts (Eds.), Building Performance Simulation for Design and Operation, 2011, Routledge, UK, ISBN: 978-0-415-47414-6, LBNL Technical Report LBNL-4792E.pdf.
[14] S. Petersen, S. Svendsen, L.D. Christoffersen, Simulation-based support for integrated design of new low-energy office buildings (Ph.D. thesis), Technical University of Denmark, Department of Civil Engineering, Kgs. Lyngby, 2011, DTU Civil Engineering Reports, no. R-247.
[15] C.J. Hopfe, J.L.M. Hensen, Uncertainty analysis in building performance simulation for design support, Energy Build. 43 (2011) 2798–2805.
[16] C.J. Hopfe, M.T.M. Emmerich, R. Marijt, J. Hensen, Robust multi-criteria design optimization in Building design, in: Proc. First IBPSA—Engl. Builiding Perform. Simul. Assoc. BSO12, 2012, pp. 118–125.
[17] K. Negendahl, T. Perkov, A. Heller, Approaching Sentient Building Performance Simulation Systems, in: E.M. Thompson (Ed.), Proc. eCAADe, Newcastle, UK, 2014, pp. 1–11.
[18] M. Wetter, E. Polak, A convergent optimization method using pattern search algorithms with adaptive precision simulation, Build. Serv. Eng. Res. Technol. 25 (2004) 327.
[19] J.A. Wright, A. Brownlee, M.M. Mourshed, M. Wang, Multi-objective optimization of cellular fenestration by an evolutionary algorithm, J. Build. Perform. Simul. 7 (2014) 33–51.
[20] L. Caldas, An Evolution-based Generative Design System: Using Adaptation to Shape Architectural Form, Massachusetts Institute of Technology, 2001, http://dspace.mit.edu/handle/1721.1/8188?show=full.
[21] W. Wang, R. Zmeureanu, Two-phase application of multi-objective genetic algorithms in green building design, in: Ninth Int. IBPAS Conf., 2005, pp. 1323–1330.
[22] J. Kim, Y.K. Yi, A.M. Malkawi, Building Form Optimization in Early Design Stage to Reduce Adverse Wind Condition—using Computational Fluid Dynamics, in: 12th Conf. Int. Build. Perform. Simul. Assoc., Sydney, Australia, 2011, pp. 14–16.
[23] J.D. Gerber, S.-H. Lin, Geometric complexity and energy simulation, in: B.T.R. Stouffs, P. Janssen, S. Roudavski (Eds.), Proc. 18th Int. Conf. Comput. Archit. Des. Res. Asia (CAADRIA 2013), Hong Kong, Singapore, 2013, pp. 87–96.
[24] D. Gerber, Designing-in performance through parameterisation, automation, and evolutionary algorithms: 'H.D.S. BEAGLE 1.0, CADRIA 2012, in: Proceedings of the 17th International Conference on Computer Aided Architectural Design Research in Asia/Chennai, 25–28 April 2012, 2012, pp. 141–150.
[25] M. Turrin, P. von Buelow, A. Kilian, R. Stouffs, Performative skins for passive climatic comfort, Autom. Constr. 22 (2012) 36–50.

[26] P. Sanguinetti, M. Bernal, M. El-khaldi, M. Erwin, Real-time design feedback: coupling performance-knowledge with design iteration for decision-making, in: Proceedings of the 2010 Spring Simulation Multiconference, SpringSim 2010, Orlando, Florida, USA, April 11–15, 2010, http://dx.doi.org/10.1145/1878537.1878737
[27] J.A. Sargent, J. Niemasz, C.F. Reinhart, SHADERADE: combining rhinoceros and EnergyPlus for the design of static exterior shading devices, in: 12th Conf Int. Build. Perform. Simul. Assoc., 2011, pp. 14–16.
[28] M.T. Wu, Z.Y. Wang, Q. Butala, S. Darwin, Optimization Framework User Manual, Bentley Systems Incorporated, Watertown, CT 06795, USA, 2011.
[29] D. Rutten, Evolutionary principles applied to problem solving using galapagos, in: AAG10, Vienna, 2010.
[30] M. Roudsari, C. Mackey, A. Yezioro, C.S. Harriman, P. Chopson, S. Ahuja, Honeybee, 2014, ⟨https://github.com/mostaphaRoudsari/Honeybee⟩.
[31] K. Negendahl, Termite, 2014, ⟨http://cobalab.dk/2014/07/15/termite-1-0/⟩.
[32] ISO EN ISO13790, Energy Performance of Buildings—Calculation of Energy Use for Space Heating and Cooling, 2008.
[33] J.E. Christensen, P. Schiønning, E. Dethlefsen, Comparison of simplified and advanced building simulation tool with measured data, in: 13th Conf. Inernational Build. Perform. Simul. Assoc., Chambéry, France, 2013, pp. 2357–2364.
[34] M. Benedikt, To take hold of space: isovists and isovist fields, Environ. Plann. B 6 (1979) 47–65.
[35] H. Yi, Y.K. Yi, Performance based architectural design optimization: automated 3D space layout using simulated annealing, in: Build. Simul. Conf. 2014 ASHRAE/IBPSA-USA, Atlanta, GA, 2014, pp. 292–299.
[36] F. Ascione, N. Bianco, C. De Stasio, G.M. Mauro, G.P. Vanoli, A new methodology for cost-optimal analysis by means of the multi-objective optimization of building energy performance, Energy Build. 88 (2014) 78–90.
[37] G. Zwettler, P. Track, F. Waschaurek, R. Woschitz, E. Hagmann, S. Hinterholzer, Automated building construction design optimization for reduction of construction costs and energy demand, in: Comput. Aided Syst. Theory—EUROCAST 2011, 2012, pp. 216–223.
[38] A.M. Malkawi, Developments in environmental performance simulation, Autom. Constr. 13 (2004) 437–445.
[39] J.A. Wright, H.A. Loosemore, R. Farmani, Optimization of building thermal design and control by multi-criterion genetic algorithm, Energy Build. 34 (2002) 959–972.
[40] J. Harding, S. Joyce, P. Shepherd, C. Williams, Thinking topologically at early stage parametric design, in: Adv. Archit. Geom., Springer, Vienna, 2012, pp. 67–76.
[41] B. Raphael, Multi-criteria decision making for collaborative design optimization of buildings, Built Environ. Proj. Asset Manage. 1 (2011) 122–136.
[42] W. Wang, R. Zmeureanu, H. Rivard, Applying multi-objective genetic algorithms in green building design optimization, Build. Environ. 40 (2005) 1512–1520.
[43] B.T.R. Stouffs, P. Janssen, S. Roudavski, A design method for multicriteria optimization of low exergy architecture, in: CAADRIA 2013, 2013, pp. 117–126.
[44] D.E. Grierson, Pareto multi-criteria decision making, Adv. Eng. Inf. 22 (2008) 371–384.
[45] R. Vierlinger, Octopus, 2014, ⟨http://www.grasshopper3d.com/group/octopus⟩.
[46] E. Zitzler, M. Laumanns, L. Thiele, SPEA2: Improving the Strength Pareto Evolutionary Algorithm, 2001.
[47] Danish Building Regulations, Lavenergiramme for kontorer, skoler, institution m.m. Section 7.2.4.2, 2013, ⟨http://bygningsreglementet.dk/br10_03_id161/0/42⟩.
[48] Wolfram, ERF, http://mathworld.wolfram.com/Erf.html. (2014).
[49] C. Bleil De Souza, I. Knight, Thermal performance simulation from an architectural design viewpoint, in: Presented at: Sixth International IBPSA Conference, China, 2007, pp. 87–94.
[50] K. Negendahl, Parametric design and analysis framework with integrated dynamic models, in: Proc. Third Int. Work. Des. Civ. Environ. Eng., 2014.
[51] R. Perez, P. Ineichen, R. Seals, J. Michalsky, R. Stewart, Modeling daylight availability and irradiance components from direct and global irradiance, Sol. Energy 44 (1990) 271–289.

Parametric design and analysis framework with integrated dynamic models

Kristoffer Negendahl[1]
krnj@byg.dtu.dk
[1] Department of Civil Engineering, Technical University of Denmark, Denmark
[1] Section for Building Energy, Grontmij, Denmark

Abstract: In the wake of uncompromising requirements on building performance and current emphasis on sustainability, including building energy and indoor environment, designing buildings involves elements of expertise of multiple disciplines. However, building performance analyses, including those of building energy and indoor environment, are generally confined late in the design process. Consequence based design is a framework intended the early design stage. It involves interdisciplinary expertise that secures validity and quality assurance with a simulationist while sustaining autonomous control with the building designer. Consequence based design is defined by the specific use of integrated dynamic modeling, which includes the parametric capabilities of a scripting tool and building simulation features of a building performance simulation tool. The framework can lead to enhanced awareness of building performance in the early stages of building design, thus improving energy performance and many other quantifiable performance objectives.

Keywords: Integrated Dynamic Model, Consequence Based Design, Parametric tool, Building Performance Simulation, Integrated design.

Introduction

Near 80% of the design decisions impacting energy consumption are made during the first 20 % of the design process (Theßeling, Schlüter, & Leibundgut, 2008). Regardless of the numerous attempts to structure the design process, and in so doing improve performance of buildings, no tangible multi-disciplinary structure is found in the design approaches in Danish building design practices today. One reason is the diverse and segregated culture differences between those who design and those who calculate (Bleil de Souza, 2012). Another reason is the absence of tools that meet the performance analysis needs of both architects and engineers in early design (Toth et al., 2011).

Many engineering consultancies offer architects building energy consulting expertise in the early stages of building design, but very few projects are fashioned within a true *integrated design process*[1]. This means only few projects today are designed by *design teams* consisting of experts in many disciplinary fields of the AEC industry.
Energy analysis (and other performance analyses and assistance) are for this reason either handled by the architects themselves or not considered at all. Engineers usually assist the architect with the aid of building performance simulation (BPS) tools in later stages, which often results in easy-fix solutions which are far from ideal in terms of performance, cost-efficiency and the overall holistic and human centered solutions.

[1] Integrated design as defined by IEA task 23 (Löhnert et al., 2003)

Background

The analysis procedure when operating BPS tools requires a user with suitable knowledge of the tools and understanding of building physics as well as insight in regulatory building energy requirements. Of this reason BPS tools are often handled by experts (often associated with a simulationist or engineering analyst (de Souza, 2009)). Nevertheless different software developers (e.g. IES (Integrated Environmental Solutions, 2013), Autodesk (Autodesk, 2013c)) have over the past few years produced simplified versions of their BPS products to accommodate building designers, thus making the process of energy simulation an accessible task for non-simulationists. The simpler tools have been proved very powerful in improving energy performance from earliest design stages which is demonstrated in previous studies e.g. (Bambardekar & Poerschke, 2009; Doelling & Nasrollahi, 2012). However non-simulation-experts may have difficulties establishing the required *quality assurances* (Hensen, 2004) when handling the BPS environments, meaning non-simulationist often do not have the required competences to use and analyze BPS. The inclusion of a specialist in the early design stage is paramount to achieve a valid ground for informed design.

The geometric modeling procedure when using BPS tools has been reduced dramatically due to better interoperability between BPS tools and *design tools* (often classified as CAD software). The technical foundation for model level collaboration has of this reason improved significantly. Yet many problems still exist when seeking to either unify or couple design tools and BPS tools (Negendahl, 2015). One solution in coupling the design tool and the BPS tools is by introducing a middleware. A scripting tool can act both as middleware while it can enhance modeling prospects by integrating parametric variables into the model. In the past few years scripting tools have become more common amongst engineers and architects. Even though scripting tools (e.g. Grasshopper (Robert McNeel & Associates, 2013a), GenerativeComponents (Bentley, 2013), Dynamo (Autodesk, 2013a) and Design Script (Autodesk, 2014)) are used very differently by architects and engineers, the ecosystem of parametric tools and scripting environments are gradually changing the way architects and engineers cooperate. Toth et al. (Toth et al., 2011) were the first to suggest the combined use of scripting tools and BPS tools in a collaborative environment. With the aid of a plug-in structure to link a design tool e.g. Rhino (Robert McNeel & Associates, 2013b) or Revit (Autodesk, 2013c) with a BPS tool it is possible for the building designer to maintain control of the geometric properties of a model and the BPS in the same unified modeling environment. Yet again the quality assurances mentioned by Hensen (Hensen, 2004) are still missing when using these parametric building performance simulation models.

Three major challenges in using BPS in building design are identified:

1) Building designers are rarely experts in building performance evaluations.

2) Expert knowledge in building performance is rarely part of the early design stages.

3) Integrated design methods in the early design stages are a rare phenomenon.

Introducing consequence based design

Consequence Based Design is an interdisciplinary dynamic framework for improving building performance[2] in the early design stage.
The framework can support of any type of analytical performance metric and is able to provide consequence feedback of design changes to a building designer in any stage of the design process. This article focuses on the use of consequence based design as a framework to improve building energy and indoor environment in the early design stage.
Consequence based design provides visual building performance simulation feedback in the building designer's native design tool (CAD) while maintaining the validity and accountability provided by a simulationist. The framework is defined by the use of three categories of tools operated by three categories of operators; together creating and employing an *integrated dynamic model:* a combination of: 1) Design tool 2) VPL (visual programming language / scripting tool) and 3) BPS environment. (Negendahl, 2015)
Each of these tools is operated by a specialist, hence introducing a new role distribution challenging the classic separation of the architect and the engineer. This new role distribution is based on personal expertise rather than disciplinary background, focusing on three subjects 1) building (model) design 2) building model parameterization 3) building model analysis as shown in Figure 1

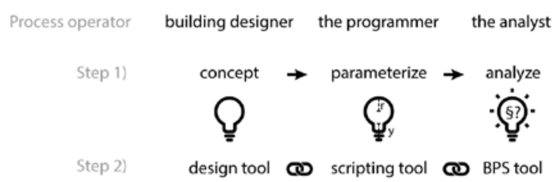


Figure 1. Consequence based design framework.
Step 1 – creating and defining an integrated dynamic model. Step 2 – operating the integrated dynamic model.

The design process is defined by two steps: step 1 is defined by the creation and definition of an integrated dynamic model and step 2 is the operation of the integrated dynamic model. Step 1 is about forming the scope of the design while step 2 is about the exploration of the design.

[2] Building performance is in this article refereeing to calculable, measurable performances such as energy consumption, temperature, cost, etc.

Integration and collaboration in the early design stages

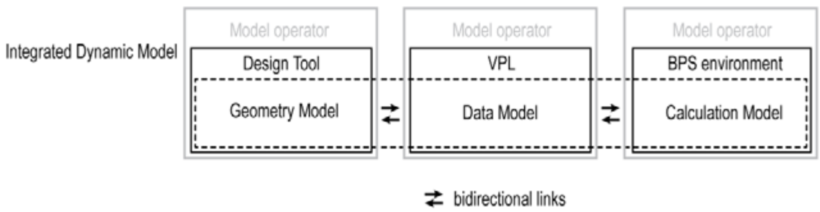
Integrated dynamic models can be operated by anyone (see Figure 2), which is why such model can accommodate the building designer alone as characterized by the simplified standalone BPS tools such as Vasari (Autodesk, 2013d) and Ecotect (Autodesk, 2013b). The real difference from these standalone tools is the option to operate the integrated dynamic model in a collaborative and highly custom environment, of the simple reason that the tools defining an integrated dynamic model remain separated. Additionally, the ability for the user to customize the design environment to fit the special needs of the particular operators, is an exclusive characteristic of the integrated dynamic model (Negendahl, 2015).

The process of creating and using an integrated dynamic model is somewhat similar to the integrated design process as described by Löhnert et al. (Löhnert, Dalkowski, & Sutter, 2003) and utilized by e.g. Petersen (Petersen, 2011). The method detaches itself from the traditional integrated design process by being able to support changing criteria and multiple parallel concepts, using advanced parametrical operations and is defined by the integration of runtime linked BPS tools. When the model is created the method allows the building designer to work intuitively and uninterrupted with building performance feedbacks. The framework embraces the supremacy of "computer intellect" in terms of calculation, data analysis, and information retrieval while acknowledging the superiority of human intelligence when it comes to strategy of designing high performance buildings. The integrated dynamic model is introducing a whole new level of disciplinary independence that may open up for new creative solutions and more focus on the analytical part of design exploration.

In contrast to other goal seeking methods based on integrated design processes, the framework of consequence based design supports the experimenting nature of architecture in the early design stages and is highly adaptable in terms of BPS tools, consequence feedbacks as well as performance representation forms. The framework describes a *data driven approach* rather than a *criteria driven approach*, meaning the method is applicable outside the realms of structured design teams and role definitions often required in integrated design.

The framework of consequence based design is developed to improve the support of performance expertise in the early design stage by acknowledging the simple train of thought; conflicts between people in the early design stages lead to poor interdisciplinary collaboration, poor collaboration results in poorly designed buildings. When conflicts arises around the matter *of who is in control* and *who is the decision maker* when determining and crafting buildings, a design team is ineffective, since the conflicts may very well rise between the parties in the design team.
If a team of experts is to design high performance buildings in a collaborative environment, such an environment must be very attentive to cultural, ideological and basic human differences. Consequence based design is dealing with this delicate matter by utilizing *the model* as a medium to distribute knowledge.

The idea is to define custom integrated dynamic models aimed for the building designer, thus maintaining the building designer as the lead operator role in the early design stage. The model is created with the *intent of experimentation.* And as a fundamental part of an integrated dynamic model is the presence of parametric objects and variables. The consequence feedback from the linked BPS tool(s) is

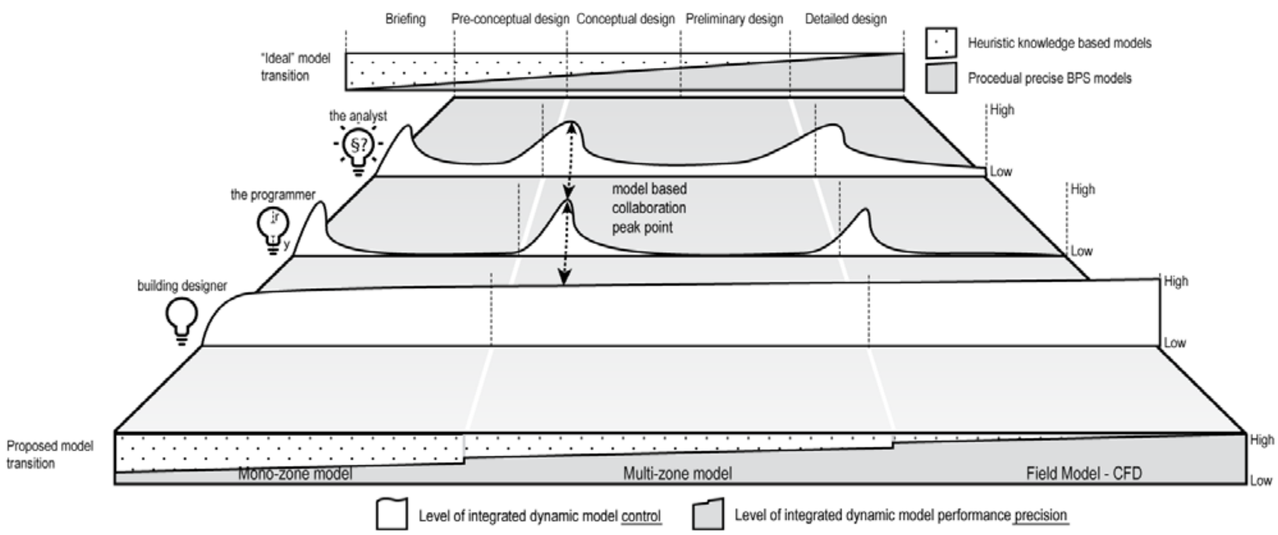


translated into direct visual input within the design tool, thus providing valid results in an integrated environment.

The consequence based design framework seeks to answer the lack of operator expertise by defining a division of operator-to-tool solution, thus shifting the avertable interaction between people to an interaction through models. Bear in mind, when the model is created, the BPS tool is linked to the design tool through a scripting tool, this combination of tools gives the simulationist the option to define the necessary input requirements of a building performance simulation, in this framework *embedded expertise* (Kilian, 2006) arises. Following embedded expertise, the building designer is less depended on operator-to-operator interaction, therefore expanding the autonomy of the designer, which many architects' requests (Banke, 2013; Hermund, 2012). It is likely that an adequate amount of embedded expertise, when provided by a combination of a geometric model and results from a calculation model will deliver sufficient knowledge to make up for the lack of knowledge held by the practitioner. However, embedded expertise when provided from a computer simulation, does not necessarily ensure sufficient analytical experience of the practitioner (Negendahl, 2015). For this reason, the framework of consequence based design is sustaining the analytical responsibility with the simulationist. The role of the simulationist is to become all the things the simulation tool cannot be. By using integrated dynamic models the simulationist is able to automate rules of input variables, requirements of systems and even decision sequences, thus making the job of the simulationist less focused on performance simulations and more on performance analysis.

The dynamics of the design process

Shaviv et al. (Shaviv, Kalay, & Peleg, 1992) has described an ideal transition from heuristic knowledge into the use of various procedural precise calculation models (See upper part of Figure 3). The argument is that some BPS models are more suitable than others in supporting the building designer in the various decompositions of the design stages. This means, to better support the designer, the BPS tool has to follow the information level throughout the dynamics of the design process. While various standalone BPS tools can effectively be used to calculate the impact of design changes throughout the early design process, the very fact that multiple models has to be built in multiple environments may be a problem for the designer. Also the requirement of multiple tools further reduces performance quality assurances as mentioned by Hensen (Hensen, 2004). An integrated dynamic model support BPS tools at any scale and any information level, and the framework of consequence based design follows the idea of using procedural precise BPS tools throughout the design process. Figure 3 shows the use of various types of integrated dynamic models,



where the linked BPS tools are replaced as the design decisions becomes more stable and more data has been collected. The figure also depicts the shifting control level of the integrated dynamic model. Every time a new type of analysis is required, the model must accommodate new input, new variables and parametric definitions. The model control peak points of the programmer are defined by the new implementation of parametric relationships and changes in the code. The model control spikes with the analyst are representing the necessary implementation of requirements to perform meaningful simulations with of the changing model. To make sure the new model works as intended, the building designer, the programmer and the analyst must collaborate. In the figure (Figure 3), the peak points of collaboration at model level can be seen when the three operators level of model control overlap. Ideally, collaboration between the programmer and analyst is initiated before a new model is taken into use by the building designer. In this way, many BPS tool specific requirements can be integrated before the model requirements and terms of experimentation from the building designer is implemented, leaving more time to implement the building designer's specific requests of the model. The collaboration between the programmer and the building designer is required to provide sufficient variability in the parametric definitions. Since the analytic knowledge (embedded expertise) of performance is distributed through and by the model, the contact between the building designer and the analyst is fully confined to the model. However, collaboration in the human interaction domain will always be beneficial for the process.

Compared to the integrated design methodology, the consequence based design framework only requires few peak points of collaborational interaction between the operators. The process of integrated design is often associated with parallel operators and team decisioning in continuous iterating loops. The consequence based design framework does not require continuous human interaction. Once the integrated dynamic model is created, the programmer will only assist in using and modifying the model, the analyst will only monitor and provide feedback of the issues the model itself cannot handle, while the building designer have the actual model control. As seen in Figure 3 the building designer remains in control with the model throughout the entire process. Even if fairly advanced simulation tools such as CFD are used, the building designer will never notice the difference in analysis complexity. The analyst remains in charge of providing valid input variables as well as managing important performance results back to the design tool. As Kaley et al. (Shaviv et al., 1992) advocated with their procedural precise models, it is suggested to use simple mono-zone models in the earliest stages, transit into multi-zone-models and in the end, when assumptions are unnecessary, field models can be initiated.

The true value of using the consequence based design framework over the integrated design framework is the acknowledgement that building performance consists of more than a result from a BPS tool. The consequence based design handles unquantifiable parameters and objectives which are equally important as of those that are measurable and calculated. Consequence based design is about being aware of the consequences of design choices, since performance feedback is only one of many consequences the building designer should react upon. By integrating the performance feedback into the design tool, the designer has both the visual feedback of the geometry and the performance in the same place, thus aligning *performance* with *composition*, *layout*, *aesthetics*, *social impact* and many other unquantifiable characters of a building.

Discussion

Daniel et al. (Daniel, Jane, & Mark, 2011) observed that if parametric modelling is to become central to the design process, then it will be necessary to deal with complexity and particularly in a collaborative environment.

With the introduction of the operator; the programmer is removing the accountability of a coding-skilled building designer. However, a skilled building designer may very well take the role as the programmer, thus advancing the parametric possibilities of the framework even further.
The real challenge in using the integrated dynamic models in integrated environments is how criteria and goal specification are understood by the design team. The way the team collectively handles the few collaboration points throughout design process (as seen in Figure 3) is crucial of how well the integrated dynamic model is *integrated* into the design.

Undefined, inaccurately followed and misinterpreted performance requirements can in worst case lead to a poorly performing building design. Of this reason, when building performance is an important aspect of the building design, clear performance objectives are required. Following that statement, every requirement must be acknowledged by all operators of the model.
Bachman (Bachman, 2004) described his concern of the role as a building designer compromised by *integration;* if building designers still wish to control their design as a whole instead of purely becoming *professional specifiers*, the integration of requirements must be handled with care. Integration

can be a dangerously all-inclusive term. Once the idea of integration is announced, any conversation on how to attain it can easily become unduly elaborate and all-encompassing. The problem is one of scope: *Integration is about bringing all of the building components together in a sympathetic way and emphasizing the synergy of the parts without compromising the integrity of the pieces* (Bachman, 2004).

In this respect, the idea of consequence based design is simply emphasizing the consequences of design choices in terms of particular performance metrics. The integration is not about people, nor about requirements, but knowledge and information. Performance feedback from an integrated dynamic model is itself a valuable piece of information, in which the building designer can chose to react upon. While the desire of the building designer in conforming to certain (performance) requirements are not explicitly necessary, the framework of consequence based design is defined by visualizing (or by other means provide) the performance consequences from the designers own choices. In this way the building designer needs to behave towards the responsibilities following choices he or she makes.

Such behavior can lead to many interesting directions. The first reaction of the building designer may be experimentation of design solutions based on own intuitions of performance. The intuition is then challenged by the hard data of the performance feedback. This process can change the perception and intuition of non-simulationist to require better understanding of the buildings they design. In this way consequence based design is a very un-intrusive framework to support performance based design, however it will never guarantee better building performance. Only the building designer reactions to the consequences of their design choices can lead to better performing buildings.

Another use of the framework is the building designer acting as a problem solver. The problem solver will seek out to minimize energy consumption, improve indoor environment, or whatever the scope is in terms of the provided performance feedback. In such cases, the quality of the integrated dynamic model is the only limiting factor to what extent the building performance can be optimized.

Consequence based design is defined by the explicit use of integrated dynamic models; this means that the BPS tool can cover any simulation tool, which may be runtime coupled with a scripting tool. Thus, introducing multidisciplinary collaboration and true awareness on subjects which require expert knowledge e.g. structural performance, life cycle cost, environmental and social impact. In any of these cases, a coordinated goal specification between the analyst and the building designer can lead to high performance buildings. This is of course optional, but the framework supports a much more objective focused design method if the team wishes so. With clear objectives, advanced automation features such as multi criteria optimization, shape grammar and agent based assistants can be incorporated into the integrated dynamic model without great difficulties (Negendahl, 2015). The middleware, the scripting tool, is the sole reason of an easy model transition with these options. However, in such situations, the term “optimization” has to be redefined to accommodate the enhancement of both the quantifiable and the quality defined objectives which is yet to be resolved.

Conclusion

In the pursuit of improving building performance, the way experts from the various disciplines in the AEC industry are collaborating is redefined by the framework of consequence based design. While building designers rarely are experts in building performance evaluations, the parametric framework of consequence based design puts the building designer in charge of one or more BPS tools. Only through an integrated dynamic model is the validity and required expert knowledge present without continuous attendance of a simulationist. This gives the building designer a new level of freedom and autonomy to experiment with the building design, while relying on valid performance feedback. The framework gives the simulationist more time to analyze results and support the building designer in ways a BPS tool cannot, meaning the quality and depth of the building performance analysis can be greatly improved. Tedious tasks of reminding the building designer to adjust window-wall-ratios and using sufficient insulation are no longer necessary. However, with the actual presence of a simulationist acting as an expert analyst, knowledge in building performance has become a central part of the early design stages.

While consequence based design detaches itself from many procedural ideas in the integrated design process, the framework of consequence based design is capable of delivering the same end goals as described in IEA task 23 (Löhnert et al., 2003). However, consequence based design is not explicit distinct by improving energy performance and indoor environment, since any simulated performance metric by a can be used. Treating quality defined and quantifiable objectives as equal is what makes consequence based design a framework superior in the integration of disciplinary expertise in the early design stage.

Autodesk. (2013a). Dynamo. Retrieved from http://wikihelp.autodesk.com/Vasari/enu/Community/Work_in_Progress/Dynamo_for_Vasari

Autodesk. (2013b). Ecotect. Autodesk. Retrieved from http://usa.autodesk.com/ecotect-analysis/

Autodesk. (2013c). Revit. Autodesk. Retrieved from http://www.autodesk.dk/revit

Autodesk. (2013d). Vasari. Autodesk.

Autodesk. (2014). Design Script. Retrieved from http://designscript.ning.com/

Bachman, L. (2004). *Integrated buildings: the systems basis of architecture*. John Wiley & Sons, Inc.,Hoboken, New Jersey. Retrieved from http://books.google.com/books?hl=en&lr=&id=9lLtElb6hlsC&oi=fnd&pg=PR5&dq=integrated+buildings+the+systems+basis+of+architecture&ots=1Bz6zZ7E79&sig=3z1Bns0xdMDArzwhul3AkNYSjAc

Bambardekar, S., & Poerschke, U. (2009). The architect as performer of energy simulation in the early design stage. In *Proceedings of the 11 th International IBPSA Conference. Glasgow, Scotland* (pp. 1306–1313). Retrieved from http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:THE+ARCHITECT+AS+PERFORMER+OF+ENERGY+SIMULATION+IN+THE+EARLY+DESIGN+STAGE#0

Banke, T. (2013). *Parametri i praksis - Generativ performance i arkitektur*. Det Kongelige Danske Kunstakademis Skoler for Arkitektur, Design og Konservering.

Bentley. (2013). GenerativeComponents. Bentley. Retrieved from http://www.bentley.com/GetGC

Bleil de Souza, C. (2012). Contrasting paradigms of design thinking: The building thermal simulation tool user vs. the building designer. *Automation in Construction*, *22*, 112–122. doi:10.1016/j.autcon.2011.09.008

Daniel, D., Jane, B., & Mark, B. (2011). Untangling Parametric Schemata : Enhancing Collaboration through Modular Programming. *CAAD Futures 2011*.

De Souza, C. B. (2009). A critical and theoretical analysis of current proposals for integrating building thermal simulation tools into the building design process. *Journal of Building Performance Simulation*, *2*(4), 283–297. doi:10.1080/19401490903349601

Doelling, M., & Nasrollahi, F. (2012). Building Performance Modeling in Non-Simplified Architectural Design. *Proceedings of the 30th CAAD Curricullum*, *1*, 97–106. Retrieved from http://cumincad.architexturez.net/system/files/pdf/ecaade2012_116.content.04662.pdf

Hensen, J. L. M. (2004). Towards more effective use of building performance simulation in design. *Proceedings of 7th International Conference on Design & Decision Support Systems In Architure and Urban Planing*, 2–5.

Hermund, A. (2012). *Anvendt 3D modellering og parametrisk formgivning*. *Design*. Det Kongelige Danske Kunstakademis Skoler for Arkitektur, Design og Konservering.

Integrated Environmental Solutions. (2013). IESVE. Integrated Environmental Solutions. Retrieved from http://www.iesve.com/

Kilian, A. (2006). *Design Exploration through Bidirectional Modeling of Constraints*. Massachusetts Institute of Technology.

Löhnert, G., Dalkowski, A., & Sutter, W. (2003). Integrated Design Process Task 23 Subtask B Design Process Guidelines.

Negendahl, K. (2015). Building Performance Simulation in the early design stage: An introduction to Integrated Dynamic Models. *Automation in Construction*. doi:10.4135/9781412963947.n116

Petersen, S. (2011). *Simulation-based support for integrated design of new low-energy office buildings*. Technical University of Denmark.

Robert McNeel & Associates. (2013a). Grasshopper3d. Robert McNeel & Associates. Retrieved from http://www.grasshopper3d.com

Robert McNeel & Associates. (2013b). Rhino. Robert McNeel & Associates. Retrieved from http://www.rhino3d.com

Shaviv, E., Kalay, Y. E., & Peleg, U. J. (1992). An integrated knowledge-based and procedural system for the design of energy conscious buildings. *Automation in Construction*, *1*, 123–141.

Theßeling, F., Schlüter, A., & Leibundgut, H. (2008). Energy and Exergy Performance as Parameters in Architectural Design Sketching -a Case Study. In *eCAADe 26* (pp. 477–482).

Toth, B., Salim, F., Burry, J., Frazer, J., Drogemuller, R., & Burry, M. (2011). Energy-Oriented Design Tools for Collaboration in the Cloud. *International Journal of Architectural Computing*, *09*(04), 339–360.

Approaching Sentient Building Performance Simulation Systems

Kristoffer Negendahl[1], Thomas Perkov[2], Alfred Heller[3]
[1,2,3]Technical University of Denmark, Department of Civil Engineering
[1,2,3]http://www.dtu.dk
[1,2]{krnj|alfh}@byg.dtu.dk [3]s103520@student.dtu.dk

Building designers make decisions in early design stages that have large impact on building performance, including those of energy-, daylight- and indoor environment performance. Building performance simulation (BPS) tools can support the designer, in making better decisions, by providing the performance consequences of design choices. However BPS tools often require deep technical knowledge and is too time consuming to use to effectively support the design exploration in the early design stages. To solve this challenge, the current paper proposes: Sentient building performance simulation systems, which combine one or more high precision BPS tools to provide near instantaneous performance feedback directly in the design tool. Sentient BPS systems are essentially combining: 1) design tools, 2) parametric tools, 3) BPS tools, 4) dynamic databases 5) interpolation techniques and 6) prediction techniques as a fast and valid simulation system for the early design stage.

Keywords: *Building Performance Simulation, Parametric modelling, Visual Programming Language, Database, Responsive system, Integrated Dynamic Model*

INTRODUCTION

Human intelligence is superior in developing abstraction, creativity and imagination while computers are superior in calculation, data analysis, and information retrieval.

Let computers handle the tactics, setting up simulations of multiple solutions, analyzing results and showing the consequences, while humans handle the strategy.

This is the fundamental idea behind the game of Advanced Chess [2]. Advanced chess is a human-computer symbiotic partnership that demonstrates a human with a computer could be far superior to either a human alone or a computer alone.

While a computer is significantly more intelligent when it comes to chess tactics, a human is significantly more intelligent when it comes to strategy. This is the case for building design, yet we failed to employ the tactical skills of computers to support us in our strategy of designing buildings.

Based on this assumption, a computer supported *building performance prediction and decision making* system will be suggested. A prototype implementation of the system, focused on daylight performance feedback, is employed to explain the proposed concept of *sentient building performance simulation systems*. Due to the narrow time frame affiliated with early design stages combined with various challenges of integrating BPS tools in building design, a sentient BPS system has to be optimized by different means, which is presented and discussed in this paper.

BACKGROUND

Building design is done on basis of geometrical representations in design (CAD) tools while performance evaluation is carried out aided by building performance simulation (BPS) tools. The actual stage is to get these tools to work together in an integral system. With the introduction Building Information Modelling (BIM) and visual programming languages (VPL), the integration of design tools and BPS tools, at model level, has improved significantly (Negendahl. 2013). This tendency is strongly implemented in the design tool Rhino [8] and belonging visual programming language (VPL) Grasshopper [3]. Grasshopper coupled to a BPS (e.g. DIVA [9] and/or Energy+ [4]) have the ability of strategic scripted parameter variations in user defined models. This reduces simulation time dramatically, as each new design proposal is automatically simulated by the runtime coupled BPS tool.

Additionally the introduction of VPLs has changed the way engineers and architects think of building design. The parametric capabilities have generated everything from architectural manifests to multi criteria optimization methodologies. Most importantly the concept of parametric models is giving building designers the "tool" that matches their continuously altering idea of a building in the early design stage. Combined with BPS, VPLs are capable of assisting and informing the building designer in every thinkable building related performance.

While BPSs, VPLs and design tools are in a process of unification, the road to full integration is still far ahead. It is today highly improbable that a building designer can handle every aspect of building performance evaluation in one go: Designing with structural optimization in mind, contemplating life cycle assessments, balancing building energy with indoor environment, etc. But why shouldn't the building designer do this? Why is the designer limited to the design tool when all this "knowledge of performance" is out there to just be simulated, evaluated and taken into consideration? This article describes a way to approach some of the primary obstacles in the merging of the design tool with the BPS by describing an implementation of a simple sentient BPS system.

EXISTING SYSTEMS LACKS RESPONSIVENESS OR PRECISION

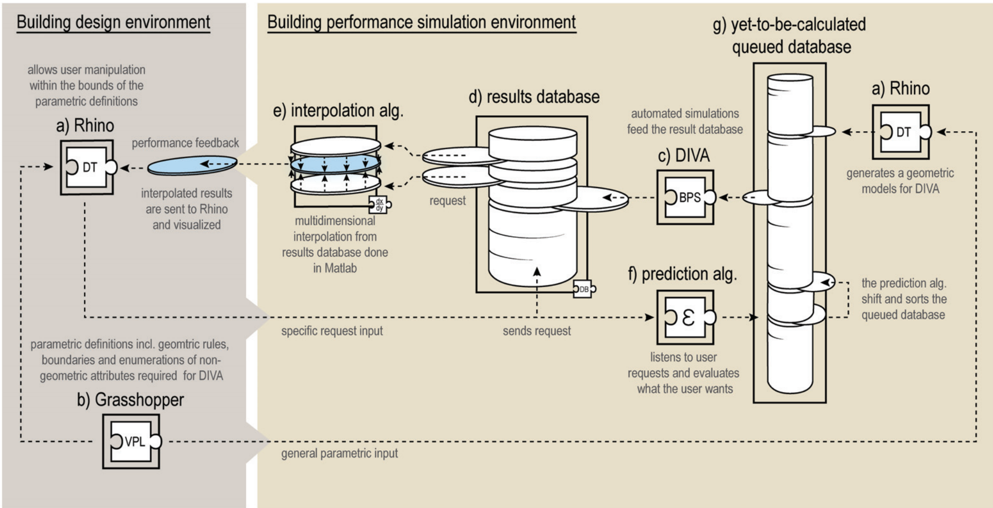
Two different approaches of linking BPS with design tools are dominating. The first approach is coupling highly detailed and complicated BPS environments to the design tools. These systems may be able to calculate the performance to a very precise degree, well beyond the information level of a building design in its conceptual stages. In general, these BPS tools needs large computing capacity and will take long time to simulate. The first approach is of this reason often much slower than the second approach and in some instances such system will block the dynamics of the design process.

The second most dominating approach seek to maximize responsiveness by either linking simplified BPS or implementing user defined scripts acting as BPS (Klitgaard et al. 2006). Ideally the right implementation and powerful computing power will allow super-responsive live performance feedback from the BPS. This approach lacks precision and may in worst case make performance evaluations on incorrect assumptions that again can lead to the very opposite of an improved building performance.

VALIDATED TOOLS, VALIDATED INPUT DATA, VALIDATED USERS

Souza (Bleil de Souza, 2012) argues that validity of modeling and calculation assumptions depends not only on the level of competency but also on the purpose of modeling. In this sense, a good model depends enormously on the experience of the modeler,

Figure 1
Implementation of a simple sentient BPS system. Rhino a) is dynamically coupled through Grasshopper b) to DIVA c) over an intermediate results database d) with interpolation features e). A prediction algorithm f) shifts and sorts yet-to-be-calculated queued data g) before the DIVA receives it.



which comes from practical knowledge and contextual understanding of the subject in order to solve similar problems. Valid operation and BPS tool input requires competent simulation experts or "simulationists" as Souza calls them.

INTRODUCING SENTIENT BUILDING PERFORMANCE SIMULATION SYSTEMS

Sentient, also meaning *consious* and *responsive* expresses an almost - human-like behavior. However interesting (and frightening) an awakening consciousness in our computer companions are, the ability of a machine to respond to building designers demands is what is important to us in this article. Sentient BPS systems are suggested as a highly responsive alternative to building designers who are either using simple proximate BPS tools or complicated but slow BPS tools in early design exploration.

The sentient BPS system is based on parametric modeling procedures, which decreases the decision space into a finite size. The system utilizes a database structure combined with a multivariate interpolation algorithm that makes it feasible to simulate less solutions and still provide the building designer with fast and precise results (from one or more building BPS tools). The idea is essentially to construct a **result database** containing building performance feedback data needed to accompany the designer's own solutions. The system further reduces the number of solutions needed to be simulated, as it observes user activity and adjusts the BPS tool to simulate and improve interpolation precision. To effectively do this, the system attempts to **predict the space of interest** of the building designer while utilizing multivariate **interpolation capabilities of the system**. Essentially the system presents building performance feedback of solutions that is of interest to the designer for decision making in the early design stages, in a very efficient way.

Sentient BPS systems are built by recognizing the need to separate the building designer and the simulationist in the early design stages. The system detaches the complexity of the BPS environment from the building designer. The building designers requests (design solutions) for performance evaluation are sent to a separated (web) performance simulation environment containing a results database, see

Figure 1. The feedback is visualized, or otherwise handled directly in the building designers' own environment , Rhino [8], in a way that fits the building designer. The simulationist will take part in the system by creating the object relations and (parametric) variables necessary to get meaningful results from the BPS. During the design process, the simulationist only job is to maintain the building performance environment (see Figure 1). The sentient BPS system can therefore support the validity of performance feedback required for any building project.

Sentient BPS systems can combine one or more high precision BPS tools and provide near instantaneous performance feedback directly in the design tool, hence providing the best of both worlds; speed and precision. The concept of sentient BPS systems is based on a further development of a student project performed by Perkov (Perkov. 2014) at the Technical University of Denmark.

THE RESULTS DATABASE

It was first suggested by Sullivan (Sullivan et al., 1988) that large number of building energy simulations saved systematically in database could provide fast feedback of energy performance. Such database is capable of giving responsive answers to multiple criteria but required either very large databases or very simple buildings to get meaningful answers. Caldas notes that these kinds of approaches generates data that do "only apply to solutions that are close to those simulated, what's makes them of limited use in an architectural design domain" (Caldas, 2001). Nonetheless, at the time of Sullivan's and Caldas' considerations were written, much development have been done in the field of databases and computing in general. It may still not be feasible to construct universal databases, comprehending every thinkable combination of variables. But it can be feasible to make a finite subset of solutions as a database lookup that takes a very specific design concept into consideration.

When constructing a database of solutions, size matters. To grasp the scale of multidimensional result databases, consider a room with three variables: its height h, its width w, and its depth d. The variables each have *variable resolution* of 10, meaning the variables may be defined in 10 different unique states. The total amount of combinations, c of solutions that need to be simulated adds up to:

$$c(r_w, r_h, r_d) = 10^3 = 1000 \text{ combinations} \quad (1)$$

If each simulation takes 5 minutes in average to simulate with a BPS tool, it will take 3.5 days to construct a result database. Now imagine we add two more variables to the equation, again each with a variable resolution of 10. We end up with almost a year of simulation time to construct the results database, which is not feasible.

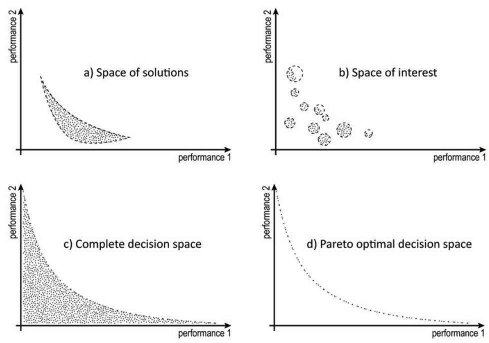


Figure 2
Decision space: Here illustrated in relation to two performance metrics, each dot represents a specific solution. Space of solutions a) is defined by all solutions that conform to the requirement of a certain performance criteria, here performance 1 and 2. Space of interest b) is defined by the building designers' interest in certain solutions related or unrelated to the performance criteria 1 and 2. The complete and pareto optimal decision spaces c), d) are shown for comparative purposes.

The solution is to limit the simulations to the problems that actually are worth investigating. Seen from a design point of view the *decision space is unlimited*. But since computer power and time are limited factors (particularly in early design stages), how and where does the designer limit the decision space?

The process of limiting the decision space can be separated into two different system approaches on designing with performance as seen in Figure 2:

- A system that supports a *space of solutions*, Figure 2a)
- A system that supports a *space of interest*, Figure 2b)

A system that finds the "space of solutions" is a system which seek to aid the designer to find solutions that complies with predefined performance criteria e.g. annual building energy consumption (Petersen, 2011). The designer usually in one way or another "pick out" a specific solution from an enumerated list of permitted solutions within the space of solutions. The system is allways limited of predefined performance criteria.

A system supporting the deductive search towards solutions: the "space of interest", allows the designer search through the solutions that may or may not comply with certain performance criteria. Typical design tools (e.g. Rhino) as well as parametric tools (e.g. Grasshopper) support deductive search of interest, usually with focus on the geometrical representation of layout, functions, visual appearances etc. Criteria of these types of qualitative objectives may be unknown to the designer until the designer suddenly *uncover* a solution that fits in a greater holistic whole. *Through a web of moves, designers discover the consequences,implications, appreciations and further moves. Within these moves, phenomena are understood, problems are solved and opportunities are exploited*. (Souza. 2012) A system based on the space of interest is therefore limited by predefined interest criteria. The real question is, when adding a BPS tool to any of the two system approaches, how does the designer use the BPS tools, or with a simulationist involved, how does the designer and the simulationist use the BPS tools in the system?

It surely should not be the BPS tool, or the assisting simulationist, that defines the design direction, but the designers own choices in *what is worth investigating*. In this regard, the sentient BPS system aligns itself with the approach of space of interest. However, sentient BPS systems may be used to narrow down the decision space by utilizing predefined performance criteria as required by the space of solutions framework. The sentient BPS system employs the parametric capabilities of a VPL to define the *space of interest*, thus narrowing down the open design problem into a smaller finite decision space. Aided by a VPL, this can be done in numerous ways:

- The building designer (and simulationist) may focus purely on an (optimization of) expected performance.
- The building designer and the simulationist may define a coordinated reduction of the decision space, seeking to advance various performance related and unrelated objectives.
- The building designer (and simulationist) may choose to setup a parametric model on the sole purpose of finding a particular desired geometric form and use the BPS results to validate the geometry as "good enough".

In the prototype sentient system discussed in this article, the simulationist and the building designer are collectively reducing the decision space by employing parametric model scripted in Grasshopper [3]. The only clearly defined objective is to improve daylight factor conditions in a room model. Other objectives such as aesthetics, layout and qualitative use of daylight is unknown to the users in the beginning of the modeling process, however, aided by the sentient BPS system the objectives becomes apparent during the design exploration with the parametric model. There are no criteria or predefined rules, others than the limitations based on the implemented parametric definitions and the parametric boundaries in the variables used in the model.

PREDICTING THE SPACE OF INTEREST

Predictions of building designer interest is a rather unexplored subject while predictions of the (space of) solutions has been thoroughly investigated e.g. by (Pedersen, 2006; Shi & Yang, 2013). Framing the space of solutions is defined by very accurately defined objectives, and in terms of building performance, the objectives have to be defined in a way that BPS tools can understand. Predicting the user interest is very different, simply because the user often does not know what he or she is interested in to begin with. The objective is an exploration in itself why

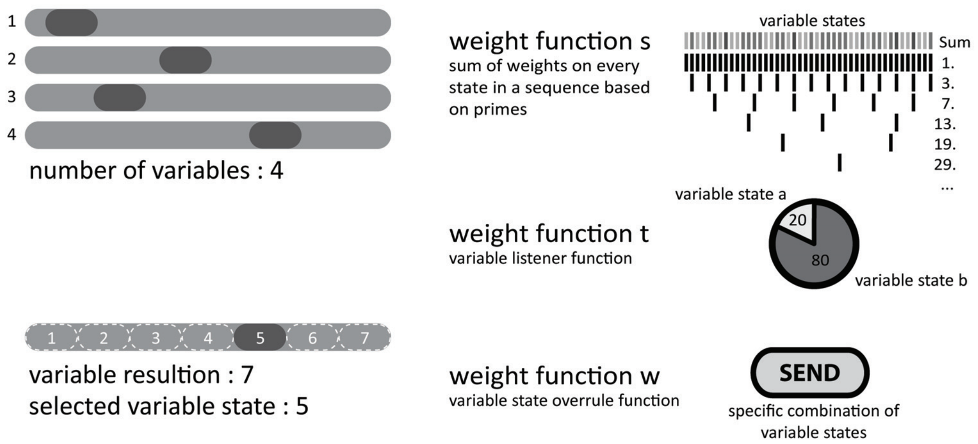


Figure 3
The prediction algorithm uses variable states (here represented on sliders). The slider in the bottom left shows a variable resolution of 7 and its current state is 5. The three weight functions are illustrated to the right.

objectives are likely to be unclear and fuzzy. The concept is to utilize embedded information of the parametric variables present in the model. The amount of variables and their *resolutions* will define the amount of unique combinations in the model as it was mentioned earlier.

A variable resolution is the amount of unique states a given (parametric) variable has. The variable is an enumeration of numbers, which does not need to be sequential or based on integers. An example is shown in Figure 3; let a variable be represented by a slider that can be set in state $[1..7]$, the variable resolution of the slider is 7. A system may have more than one slider or any other collection function of variables (navigational controllers, lists, arrays, etc.). Each individual variable has a individual variable resolution. We now introduce a concept of *variable resolution levels* to further reduce the amount of solutions needed to be simulated. The idea is to make precise performance simulations on strategically selected solutions within the space of interest, then estimate the rest of space of interest with minimum amount of errors. The *variable resolution level* of any given sentient BPS system is basically all the unique combinations of every variable states divided by the number of finished simulations (per coupled BPS tool), defined as follows:

Variable resolution level; has a number of variables $v > 1$ where each variable resolution $r > 0$, for every resolution r in the sequence i of variables v_i:

$$\frac{r_{v1} \cdot r_{v2} \cdot \ldots \cdot r_{vi}}{\sum(\text{number of simulations completed})} \qquad (2)$$

Essentially the variable resolution level indicate how much of the space of interest have been covered by simulated results. A high variable resolution level means few simulations is completed by the coupled BPS tool (in relation to the total number of potential solutions), while a variable resolution level = 1 means every possible variable state combination has been simulated. As it follows, the number of variables and their resolution will affect the variable resolution level quite substantially. An ideal model will have a minimum required number of variables each with lowest possible variable resolutions to cover the space of interest quickly in the design process. Minimizing variables and resolutions, however can be rather difficult when the building designer have not yet decided all the design objectives. Of this reason an *interest prediction algorithm* has been implemented, hence to further reduce the needed simulations, to cover the actual interest space within the boundaries of the defined variables and their resolutions.

An interest prediction algorithm is implemented on the basis of a continuous weight factorization of the yet-to-be-simulated unique data combinations. There are basically three weight functions in the prediction algorithm; s, t, w (shown in the right side of Figure 3).

Weight functions s, t and w

Weight-function, s is given to all variable states but distributed flat out semi-random by utilizing a sequence of primes. Imagine all states is distributed in a sequence, where every third state is weighted less than every fifth, every fifth is weighted less than the seventh etc. In this way the system is set in a progressive loading-state that helps to get "rough" and faster interpolations "evenly" distributed over the entire the space of interest. The idea with this function is to gradually improve the overall distribution of simulations, in the space of interest, by an incremental expansion variable resolutions.

Weight-function, t is a *variable listener function*, which essentially is a timer function that reads the particular variable states of navigational controllers (aka. sliders) embedded in Grasshopper. Basically the listener function identifies the state of every variable and how long time it remains in that state. The highest weight is given to the variable with the fewest alterations, which arguably must be the preferred state of interest of that particular variable.

Weight-function, w is a simple overwrite-function that favorably alters the weight of a given state on a given variable. It gives the building designer an option to alter a specific request (design solution) to become more important than all other requests queued for simulation. The exact combinations of variable states are sent directly to the BPS to perform an analysis based on that specific request.

weight s_j, t_j, w_j $[0..1]$, as follows: For every variable state in the sequence j :

$$\text{weight}(j) = s_j + t_j + w_j \quad (3)$$

The sum of the weights of each parameter is continuously updated while the building designer uses the system. As seen in Figure 1 g) the effect of the prediction algorithm is a reordering of the yet-to-be-calculated queued database. The consequence of using the prediction algorithm is a more efficient use of simulation power, as the requests from the building designer is automatically taken into account.

INTERPOLATION CAPABILITIES OF THE SYSTEM

Well before the results database is complete, it is possible to interpolate results by using multivariate interpolation techniques. In the prototype implementation seen in Figure 1, the choice was to use the GridDataN from the native Matlab library. GridDataN fits a hyper-surface of the form $y = F(x)$ to the data in non-uniformly-spaced vectors. When a request is sent to the interpolation algorithm it will automatically perform a linear interpolation between the continuous and sequential simulations in the results database.

To ensure that interpolation is not performed over discontinuities the user has to state which of the input variables are continuous and discrete, thus sending compliant data to the interpolation algorithm. The interpolation feature is simply a method to give fast feedback of the results "yet-to-be-simulated". While it is strictly not required in the sentient BPS system, the interpolation algorithm can give the building designer faster and more detailed feedback with fewer simulations, but it also introduces an element of uncertainty into the system.

As seen in Figure 4 the user may get fast feedback from an incomplete results database. In the prototype implementation the performance metrics is daylight factors, where interpolation was calculated over each measurement-point in a grid (648 individual points corresponds to 648 individual results). This means every time the user sends a request with a new set of variable states (in Figure 4, two dimensions, x

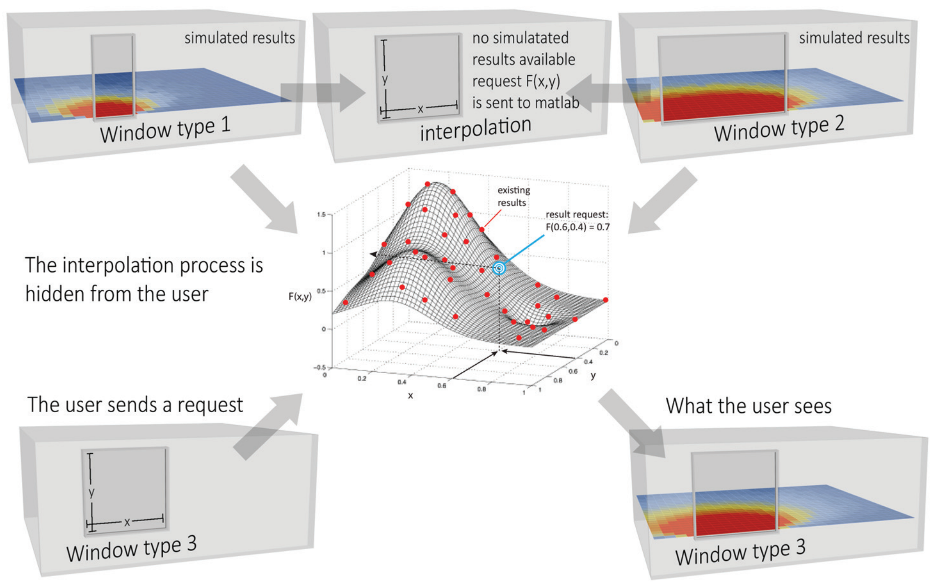


Figure 4
The interpolation process of using GridDataN, here showed with a two-dimensional interpolation F(x,y), where x is the width and y the height of a window. F represents the performance metric daylight factor [%] in 648 measurement-points (seen as colors on a plane). The red dots represent a unique product of simulation from DIVA. The blue dot represents the" lookup" procedure handled by the interpolation algorithm.

and y, representing width and height of a window), the user asks the system to look for 648 individual results from the database. If the corresponding combination of x and y is absent in the database, the system sends the request over to Matlab. GridDataN then constructs 648 individual hyper-surfaces corresponding to the number of measurement points (and not the amount of simulations already performed). Each individual red dots in Figure 3, however, are representing the separate simulations already performed of the *same measurement point in space*. The interpolations are performed 648 times from on each of these hyper-surfaces. In theory the 648 individual measurement points could be a product of 648 different BPS tools. While the system is utterly scalable, the 648 points are just used as an example to represent the daylight factor distribution in a room, the mind blowing flexibility and scalability of the system is hard to describe. However useful and interesting multivariate interpolations are, the very fact humans cannot comprehend higher levels of multidimensional operations, makes interpolations risky to use in practice.

8. DISCUSSION

When focusing on the multivariate interpolation included in the prototype sentient BPS system, the real challenge is to minimize the errors of interpolation. As with any approximate method, the utility of multivariate interpolation cannot be overextended. While various techniques exist in error minimization (e.g. Lagrange multipliers for Kriging (Vapnyarskii, 2010)), we have simply attempted to quantify the errors that can occur in multivariate interpolation. The reason of this, as it follows, is to show where to expect large errors and accordingly seek to avoid them.

In Figure 5 is shown the process of estimating the error from GridDataN in the system. The errors are calculated by subtracting the interpolated results with the actual simulated results. To simplify it further the numerical value of this difference is used to estimate the effect of simulation variable resolution levels, equation (2).

Errors of the simple case of the two-dimensional interpolation are showed in figure 6. It is seen in the upper left corner that even two exact simulations vary due to stochastic variations in simulation tool DIVA. Based on two previously discussed variables x,y, each variable has a resolution of 9, which follows a variable resolution level of: $(9 \cdot 9)/80 \approx 1$. The interpolation performs reasonably well, however errors tend to accumulate near the window, thus affecting the highest result values. The reason is associated with the inter-dimensional increasing numeric variations in the larger values, which means more simulations may be needed where changes vary much from one solution to the other. Nonetheless estimating daylight factors of 2% or below was reasonable precise even with a fraction of the simulations done.

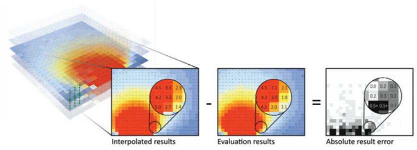


Figure 5
The absolute error of interpolation is simply the absolute value of interpolated result subtracted from the actual simulated result.

When a higher resolution level is analyzed $(9 \cdot 9)/35 = 2.25$, the errors rises accordingly. From the various test performed with the system, it was found that variable resolution levels should not exceed the level 10, since it generated too great errors to be a useful guide to the designer. Nevertheless much work still need to be done in quantifying a general assumption of error levels, number of parametric variables and variable resolutions. Generalization is further problematic when relationships between variables have strong oscillations or discontinuities. This suggests that, it may be advantageous to attempt to script additional relations between variables, particularly in the case of abstract problems for which the topology of the input and output spaces may not be clear a priori, as to produce a relationship which is as smooth as possible.

In the prototype system, the discontinuities of variables have to be identified by the user. In most cases, e.g. variables such as number of windows, type of glazing, and enumerations in general, the user can easily identify the discontinuities, however, in some cases e.g. where geometry "jumps" from a state to another, the discreteness can be difficult to identify.

While only weight-function t autodidact seeks to predict the user interest, weight-function w is more of a service for the designer to validate, or refine his or her own intuition of the building performance. The inclusion of weight-function w was found necessary in providing the user a feel of control over the model. Weight function s was found necessary to include into the model because it helped the interpolation algorithm to get enough data to make the multivariate interpolations distributed more uniformly in the vast space of interest. The function also acted as a continuous generalist refiner for the result database, thus counteracting the weight function t. If the model was left completely unattended the weight function s makes sure every combination of every variable-state is pulled through the coupled BPS tool.

The actual interesting weight-function t is incredibly simple as a concept and in its implementation. The impression that a preferred state is likely to stay the same throughout the design process is quite reasonable, however in many real situations this might not be the case.

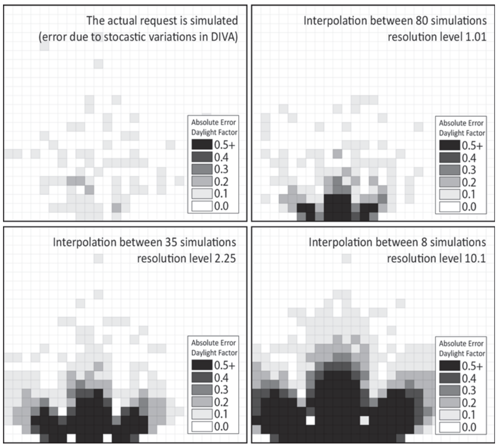


Figure 6
The absolute error of interpolation related to the resolution level (higher level means fewer simulation s available for the interpolation algorithm).

A user might not alter a variable simply because the variable is of less importance to the user. This could be the color-tone of a window pane or something else of minor interest, in terms of larger building design perspective. While this preferred state of this particular variable does not matter for the user, the system does not know the difference of inter-variable importance. This results in significantly more simulations with e.g. greenish glass, and thus giving the results of the combinations with all other variables based on this color. Greenish glass poses no physical major significance for e.g. thermal performance and building energy consumption. However, if the color green was somehow associated to the window pane g-value, the thermal performance and energy consumption will be very much affected by the choice of state. Of this reason it is suggested in future research to implement a variable importance function.

FURTHER STEPS TOWARDS SENTIENT BPS SYSTEMS

One of the most promising directions towards sentient BPS systems may be found in the fast growing field of machine learning. Of the various directions in research some popular methods are mentioned; neural networks, Gaussian processes, support vector machines, nearest neighbors, however there are many others. The way the prototype sentient system interpolates between the multidimensional results, and gradually becomes more certain over time, can be compared to many implemented machine learning concepts. However, the proposed prototype system cannot be classified as a machine learning system, of the simple reason that a hypothesis set (training data) is not a necessity for the system to work. Machine learning may very well be used to further improve the prediction of user preferred solutions, thus further narrow the the amount of simulations needed. Nonetheless, the problem with many machine learning algorithms are they often need vast amounts of data to train the system effectively (Mitchell, 2014). This needs to be addressed in future sentient BPS system builds.

Regression is concerned with modelling the relationship between variables, but unlike interpolation methods, regression does not need a continuous stream of results data to function (although, regressions are often constructed on vast amounts of empiric data). Regression can iteratively be refined by using a measure of error in the predictions made by the model. Regression methods are a work horse of statistics and have been cooped into statistical machine learning and in future sentient systems regression may be a natural next step for improved inter-dimensional estimations of building performance simulation results.

CONCLUSION

Sentient BPS systems are yet to be seen as a stable and agile implementation. Much work is needed in the area of predicting building designer requests. Better, more adaptable multivariate interpolation methods needs to be utilized. Additional features e.g. feedback of suboptimal directions will be highly beneficial for the sentience of the system. Nevertheless, the implementation of "sentient", also meaning "responsive", BPS systems promises the building designer a fast feedback with valid results.

REFERENCES

Bleil de Souza, C, BDS 2012, 'Contrasting paradigms of design thinking: The building thermal simulation tool user vs. the building designer', *Automation in Construction*, 22, pp. 112-122

Klitgaard, Jens, JK, Kirkegaard, Poul Henning, PHK and Mullins, Michael, MM 2006 'On the Integration of Digital Design and Analysis Tools', *Digital Architecture and Construction*, South Korea, pp. 187-196

Negendahl, K., KN 2013, 'Building Performance Simulation in the early design stage: An introduction to Integrated Dynamic Models', *Submitted to Automation in Construction*, ., p. .

Caldas, L., LC 2001, *An evolution-based generative design system: using adaptation to shape architectural form*, Ph.D. Thesis, Massachusetts Institute of Technology

Burry, M., MB 2013, *Scripting cultures: Architectural design and programming*, AD Primers

Sullivan, R, RS, Arasteh, D, DA, Papamichael, K., KP and McCluney, R., RMC 1988, 'An indices approach for evaluating the performance of fenestration systems in nonresidential buildings', *ASHRAE Transactions*, 94, p. .

Petersen, S., SP 2011, *Simulation-based support for integrated design of new low-energy office buildings*, Ph.D. Thesis, Technical University of Denmark.

Perkov, TP, THP 2014, *Metode til kontinuerlig præcisering af simuleringsresultater*, Master's Thesis, Technical University of Denmark.

Shi, X., XS and Yang, W., WY 2013, 'Performance-driven architectural design and optimization technique from a perspective of architects', *Automation in Construction*, 32, pp. 125-135

Pedersen, F. Ø., FØP 2006, *A method for optimizing the performance of buildings*, Ph.D. Thesis, Technical University of Denmark.

[1] www.autodesk.com/products/autodesk-simulation-family/features/robot-structural-analysis

[2] www.en.wikipedia.org/wiki/Advanced_Chess

[3] www.grasshopper3d.com

[4] www.apps1.eere.energy.gov/buildings/energyplus/energyplu about.cfm

[5] www.encyclopediaofmath.org/Lagrange_multi-pliers

[6] www.bentley.com/GetGC

[7] www.rtw.ml.cmu.edu

[8] http://www.rhino3d.com

[9] http://diva-for-rhino.com

Agent-Based Decision Control—How to Appreciate Multivariate Optimisation in Architecture

Kristoffer Negendahl, Thomas Perkov and Jakub Kolarik

Abstract

Early stage building performance optimisation as a viable approach is yet to be applied efficiently in building design processes, especially in the early design stages where the design space is open and changes are inexpensive. This article proposes a method of entire building energy optimisation in the early design stage. The main focus is to demonstrate the optimisation method, which is done in two ways. Firstly, the newly developed agent-based optimisation algorithm named Moth is tested on three different single objective search spaces. Here Moth is compared to two evolutionary algorithms. Secondly, the method is applied to a multivariate optimisation problem. The aim is specifically to demonstrate optimisation for entire building energy consumption, daylight distribution and capital cost. Based on the demonstrations Moth's ability to find local minima is discussed. It is concluded that agent-based optimisation algorithms like Moth open up for new uses of optimisation in the early design stage. With Moth the final outcome is less dependent on pre- and post-processing, and Moth allows user intervention during optimisation. Therefore, agent-based models for optimisation such as Moth can be a powerful substitute for traditional stochastic optimisation.

Introduction

Building performance optimisations during early stages of the design process are not only related to risks of high uncertainty (Lin and Gerber 2014), but also require an excessive amount of calculations that are very time consuming (Salminen et al. 2012). The early design stage can be characterized by a limited amount of information about the building's architecture and at the same time a high frequency of design changes. In contrast to that, most optimisation methods rely on clear objectives and well defined boundaries. Such requirements are rarely associated with the early design stage. The process of designing

K. Negendahl (✉) · T. Perkov · J. Kolarik
Technical University of Denmark, Kongens Lyngby, Denmark
e-mail: krnj@byg.dtu.dk

M.R. Thomsen et al. (eds.), *Modelling Behaviour*, DOI 10.1007/978-3-319-24208-8_31

can be far better described as an exploration of boundaries and objectives rather than finding the one solution within a fixed number of boundaries and objectives. For optimisation to be truly appreciated in building design both qualitative and quantitative objectives as well as “real” human control of these need to be part of the exploration process during the building optimisation.

Background

Few researchers address the complicated methodological challenges in the need to combine human evaluation with computational speeds in early stage optimisation. Nonetheless, design optimisation of building performance is increasingly being used in practice. A popular method for application of optimisation in design is based on generative models in combination with stochastic optimisation algorithms. To maintain artistic control of the optimiser (tool) many different attempts have been made. One dominating idea is to let the designer control the objectives rather than the actual geometry, for example (Caldas and Norford 2001; Gaspar-Cunha et al. 2011). Caldas and Norford (2001) argues that weights on objectives give the designer the highest level of control over the optimisation process. Caldas and Norford (2001) also notices that weighting factors to attributes of different objectives as of will to some extent constrain the design process itself. This is partially because changes of weighting factors may be discontinuous and can lead to dramatically different solutions and partially because setting up weighting factors requires a large amount of insight into all factors involved in the decision-making procedure.

Agent-based models (ABMs) used for optimisation are another method for stochastic optimisation. They are not as commonly used in building design as e.g. genetic algorithms, but ABMs find increasing interest in other design areas such as in transportation and manufacturing industries (Barbati et al. 2012). What distinguishes ABM from the more classical heuristic optimisation approaches (including stochastic and deterministic as mentioned above) is their ability to decompose a global problem into a number of smaller “local” problems that may be solved individually and simultaneously (Davidsson et al. 2007). ABMs, it should be noted, are from time to time mixed up with particle swarm algorithms (PSAs) (Liu et al. 2005). In this article we will clearly distinct the two algorithms by defining PSA-agents as individual solutions roaming in a competitive space of solutions, where ABM-agents compete in same-state solutions.

To complicate matters further the Building Performance Simulation (BPS) tools used to feed the optimisation algorithms affect the actual outcome of the optimisation process. Indeed the type, precision and quality of the BPS tool matter to great extent, but what is less obvious is that most BPS solvers approximate solutions due to adaptive variations in solver iterations (Wetter and Polak 2004) and thereby form discontinuous search spaces for the optimisation algorithm. As Wetter (2011) explains, nonlinear programming algorithms (typical stochastic optimisation algorithms) are computationally efficient (over deterministic counterpart), however they often require the cost function to be differentiable in the design parameters. Therefore, there is a conflict of the choice of optimisation algorithms preferred by designers and the tools applied in the process.

Method

The Moth Algorithm

The presented method makes use of a new adaptive, open agent-based optimisation algorithm named Moth; it has been developed by the authors. The algorithm allows parametric geometry and any other parametric variables to be controlled by individual agents. Moth stands for Multivariate Optimisation with Heterogeneous agents. It is a fully scalable algorithm, thus capable of taking any number of quantifiable objectives with any number of boundaries. Moth is a heterogeneous system; this means that every agent actually comes in opposite pairs; a minus- and a plus-agent. Moth is developed for the Rhino-Grasshopper environment (Robert McNeel & Associates 2013), and in this way it is integrated in the fast growing parametric universe supported by many enthusiasts and designers. Moth is open source and built in the IronPython programming language (Viehland et al. 2015) (Fig. 1).

Each Moth-agent has as a goal to improve its own dynamic objective. The Moth-agents can be manipulated by the model operator (designer) during the optimisation process as seen in Fig. 2. This gives the designer an option to focus on selected areas of a building (in the Euclidean space). The idea is to give the designer live decision control over a complex system that continuously seeks to find balance in the multivariate decision space. The proposed method depends on one or more building performance simulation (BPS) tools, a design tool (CAD) that can visualise the results from the BPS tools and a visual programming language (VPL) that facilitates the data shared between the BPS tools and the design tool. These kinds of models are also known as integrated dynamic models (Negendahl 2015).

The paired agents (−agent and a +agent) seek to reach the same objective by inversely changing design variables. These changes are based on the evaluation feedback from a coupled BPS tool. The evaluation of feedbacks is tied to the agent behavior through a *promotion* system. The strongest agent in the pair gets promoted by a positive feedback, thus giving the promoted agent more authority to change its design variables in its own preferred direction (either positive or negative). In this way every agent pair does not need to have any pre-defined preference or knowledge of what it is supposed to do other than what design variables it is allowed to manipulate. More details on how Moth works is found in the results Part 2.

Following Bento and Feijó's taxonomy (Bento and Feijó 1997) agents can be defined in a simple set of attributes and specifications, see Table 1.

Here, structural attributes may be physical (e.g. colour), or behavioural specifications (e.g. temperature = 35 °C, obtained from a thermal analysis). In an optimisation process, functional specifications (e.g. pleasant temperature) tend to be transformed into performance specifications defined in ranges (e.g. 18 °C < temperature

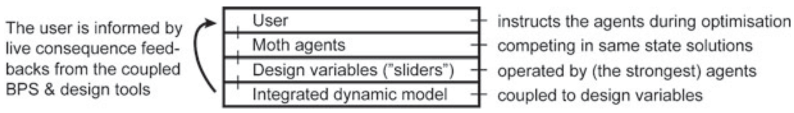


Fig. 1 Method—ABM optimization with Moth. By contrast to most stochastic optimization algorithms Moth allows changes of hyper parameters (agent instructions) during optimization

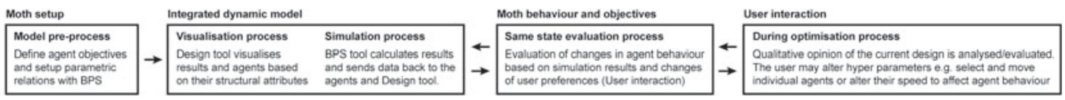


Fig. 2 MOTH—the agent-based optimisation algorithm allows user feedback and feedback from one or more coupled BPS tools facilitated through an integrated dynamic model. No post process is necessary

Table 1 Attributes of an agent-based optimisation algorithm

Type	Attribute	Representation
Structural	Colour, radius, shape	Geometrical representations
Behavioural	Temperature, daylight, energy consumption	Performance specifications

< 26 °C). Finally there are objectives describing the designer's intention for the design optimisation. The objectives may be associated, but not exclusively confined, to the performance specifications. In the following results we have demonstrated how Moth performs. The demonstration is composed in two parts:

Part 1. Moth is demonstrated on three different pre-defined single objective search spaces.
Part 2. Moth is demonstrated in a simple multi-objective case study.

Part 1

The three search spaces were tested by the Moth-algorithm as visualised in Fig. 3. The tests are compared to two other popular algorithms made available to Rhino/Grasshopper, namely Galapagos Evolutionary Solver (Rutten 2010) (the genetic algorithm, GA with default properties is used) and Goat (Simon et al. 2015) [the solver based on an controlled random search with local optimisation CRS2 (Price 1983) implemented in the NLopt nonlinear-optimisation package (Kaelo and Ali 2006)]. The purpose is not to compare different algorithms, but only to show some of the thoughts behind Moth.

The search spaces are all single objective spaces, as they are easier to visualise and compare, however all search spaces have an infinite number of "best performing solutions" thus representing an inner space of optimal solutions within the search space. The search spaces are artificially crafted to demonstrate the algorithms' ability to find optima and avoid sub-optimal solutions. These optima are represented on each search space in Fig. 3 by a dashed line, where all optimal solutions are placed exactly on the dashed line. Search space (a) is the simplest search space. It has a single large and centrally distributed circular optimal space with gradients pointing towards the optimal space in almost any point in the search space. Search space (b) is more complex and does not have any particular global gradients pointing towards the optimal space. Search space (c) is the most complex space of the three. The search space is modelled after Wetter and Polak's (2004) parametric plot of energy consumption of a building based on cooling and lighting as a function of the width of east and west facing windows. According to Wetter and Polak due to the loose solver tolerance, the cost function may exhibit "smooth" regions that are separated from each other by discontinuities. These "smooth" regions are visible as flat plateaus. The optimal spaces are placed on a very steep gradient. This type of search space is quite likely to be the type of space building designers would encounter in practice. However, in cases where multiple objectives are in play, the search space grows in dimensions and is difficult to visualise. Therefore, to demonstrate Part 2, the search space remains unknown.

Part 2

In Part 2 we applied three objectives to a test case (energy use, daylight availability and cost of windows) to a very simple building design as illustrated in Fig. 7. The building design

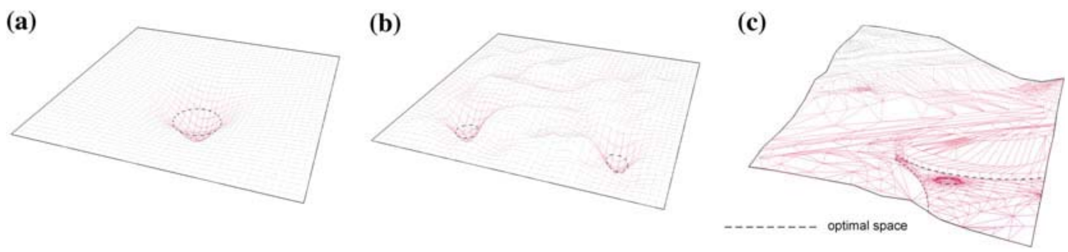


Fig. 3 Search space (**a**), (**b**) and (**c**). The optimal space is infinite solutions *on* the *dashed line*

variation for the demonstration is very limited and was controlled by three design variables. The two windows (widths) are each controlled by individual design variables and a third design variable controls window types (see Table 2). We have set three arbitrary performance specifications; daylight factor (DF %), cost of windows (−) and annual energy consumption (−) to the case. Cost indexes are defined to demonstrate the method, and do not represent real relative cost of window types.

Objective Functions

Three objective functions; $f_{energy}(x)$, $f_{cost}(x)$ and $f_{daylight}(x)$ are minimized by Moth.

$$\min_{x \in \mathbb{R}^s \times \mathbb{Z}^t} f_{daylight}(x) = O_{daylight} - nk \Big/ \left(\sum_{k=1}^{nk} (DF_x) \right) \tag{1}$$

where the fixed objective $O_{daylight}$ = 3 (%), k is the measurement point and nk is the number of measurement points in the building. DF_x is simulated by Radiance through Ladybug (Roudsari et al. 2013) for every solution, x.

$$\min_{x \in \mathbb{R}^s \times \mathbb{Z}^t} f_{cost}(x) = O_{cost} - \sum_{j=1}^{nj} \left(A_{w,j} \cdot C \right) \tag{2}$$

where the fixed objective O_{cost} = 15, j is the window and nj is the number of windows in the building, $A_{w,j}$ is the window area of the j'th window and C is the cost index shown in Table 2.

$$\min_{x \in \mathbb{R}^s \times \mathbb{Z}^t} f_{energy}(x) = O_{energy} - \sum_{i=1}^{ni} \left(Q_{heating,i} + 2.5 \cdot \left(Q_{cooling,i} + Q_{vent,i} + Q_{light,i} \right) \right) \tag{3}$$

where the fixed objective O_{energy} = 215, i is the load condition of the particular condition, ni is the number of load conditions. $f_{energy}(x)$ is simulated by Be10 (SBI 2013) through the Termite (Negendahl 2014) interface for every solution, x. The primary energy factor 2.5 is multiplied with electrical energy uses according to the Danish building regulations (2013), and the primary factor for the heat supply is set to 1.

Results

Part 1

The first comparison of the three algorithms seen in Fig. 4 shows clearly visible differences in the algorithms' approach to find minima. First, there is the "spread" of the solutions. The spread is defined by the way the algorithm distributes its solutions in the search space. One may think of this process as a search space sampling process for possible minima. Moth, compared to the two other algorithms generates less spread after a few iterations. The spread clearly only develops in a diagonally shaped direction from iteration 50 to 1000, so the first 50 iterations have a very large impact on the solutions to be explored. It almost looks like Moth already knows where to find the

Table 2 Window properties

Name	Configuration	Uwin (EN10077-1)	Ug (EN673)	G (EN410)	LT (EN4I0)	Eref (DS418)	Cost index, C
GlzTypel	4-16-4 argon	1.33	1.02	0.52	0.77	−34	1
GlzType2	4-16-4 argon	1.40	1.11	0.62	0.80	−24	1.1
GlzType3	4-16-4 argon	1.53	1.26	0.77	0.82	−11	2.6
GlzType4	4-12-4-12-4 argon	1.11	0.72	0.50	0.72	−17	1.9
GlzType5	4-12-4-12-4 argon	0.79	0.5	0.42	0.63	−2	3.6
GlzType6	4-12-4-12-4 argon	0.82	0.53	0.50	0.72	8	4.7
GlzType7	4-12-4-12-4 argon	0.90	0.62	0.62	0.73	21	6.1

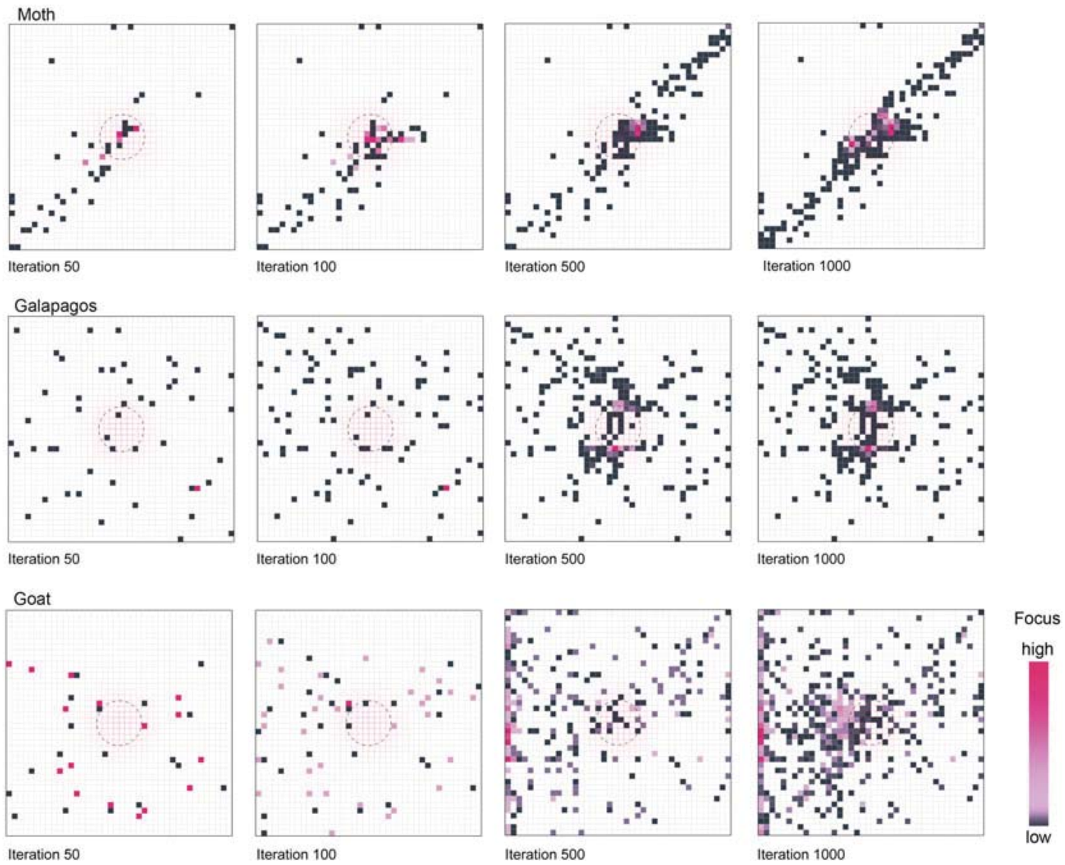


Fig. 4 Search space *A*. *Top* Moth, *middle* Galapagos, *bottom* Goat

minima. There are various explanations to this phenomenon, and this is elaborated on in the discussion.

We can also see a difference in the colouring of the solutions visited during the iterations. Dark coloured solutions have been visited once or very few times, whereas the more magenta the solutions are the more visits the areas have had. One may think of this property as the algorithm's ability to focus its search on specific areas of the search space. Here Moth is distinguished by its attention on actual minima. Around 1/3 of the circle is covered by the attention from the Moth algorithm at iteration 1000. The attention is mainly split into two clusters.

The second search space (Search space B, Fig. 5) contains two clusters of minima. The only algorithm that had success in finding both clusters was Galapagos. What distinguishes Moth from the two other algorithms is its focus and how much of the (discovered) circular optimal space it gives most attention. Moth is also much faster in homing in on one of the two clusters than any of the other algorithms. This, however, may be by chance alone given the very narrow diagonal band of spread. Again Galapagos is most evenly spread out and Goat dedicates much of its searches on the periphery of the search space.

The third search space (Search space C, Fig. 3) is by far the most difficult for optimisation algorithms to navigate in. It contains plateaus and abrupt cliff-like landscapes which arguably (Wetter 2011) are characterised by many BPS tool solver feedbacks. As illustrated in Fig. 6 Moth again finds a minimum quite fast, and again one may argue that this is based on chance, given the fact that the spread is rather limited.

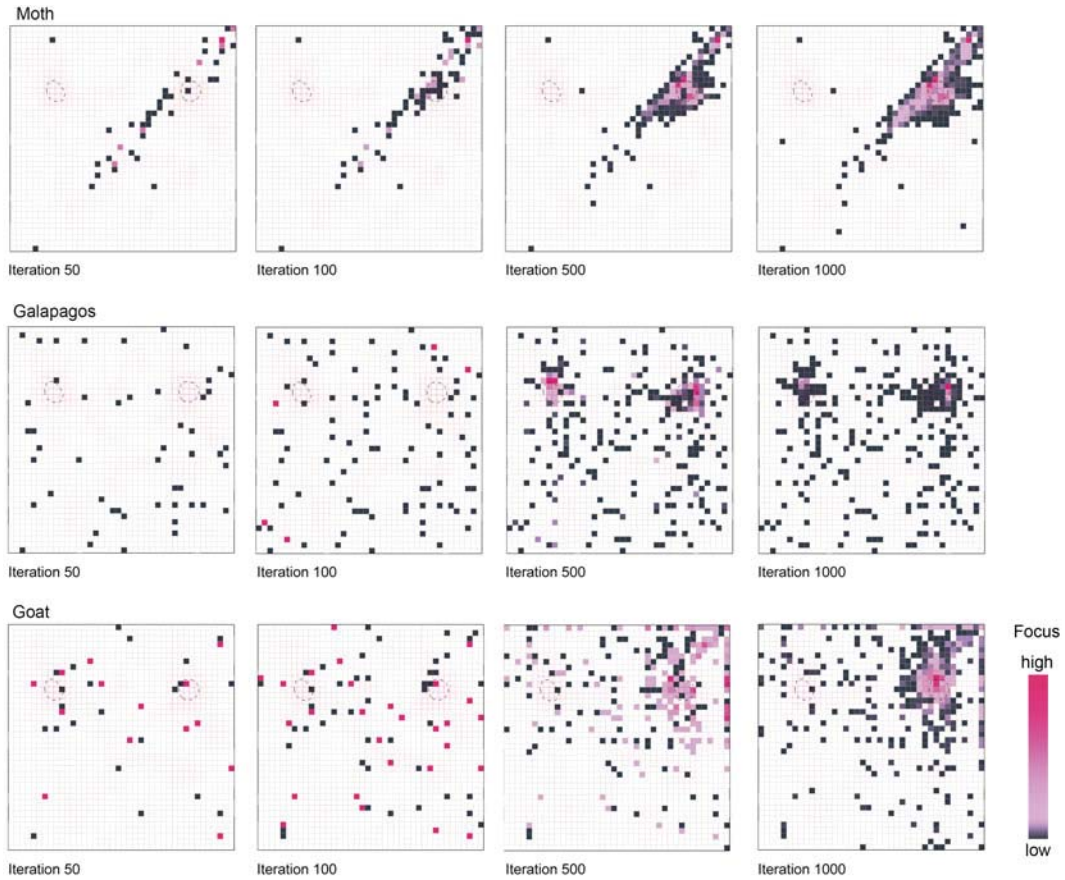


Fig. 5 Search space *B*. *Top* Moth, *middle* Galapagos, *bottom* Goat

What is interesting to notice is that the focus is not fixed upon one area but keeps moving along several minima. The last iteration reveals that the algorithm has explored about half of the optimal space and the focus is concentrated in the area that has the highest concentration of optima. Galapagos starts again with a uniform spread. The algorithm has difficulties in locating the minima and finds a near minimum plateau in the bottom right corner. What is worth noticing is that the algorithm does little to find alternatives to the point on the plateau it has *homed in* on. Goat shares the same fate as Galapagos. It has difficulties in finding the gradients that are particularly steep at the point of the optimal spaces. Interestingly, in the last iteration most of the optimal space has been uncovered by the algorithm. But its focus, similar to Galapagos, is fixed at random points on the *near minimal* plateau. And once again this will generate misleading results. Goat may be seen as the better choice for this type of search space, given its ability to uncover the widest field of near optimal space. However, Goat has not shown interest in any of the real minima spaces, so in this sense the algorithm fails to achieve what it is created to do. Moth is the only algorithm that dedicates its focus to real optima, even though the algorithm seems to settle with the left side "optimal bank" and it has not just started discovering "the island" at the latest iteration.

Part 2

The real excitement in using Moth-agents is introduced when multiple objectives and multiple agents are used in the same model. Here, various

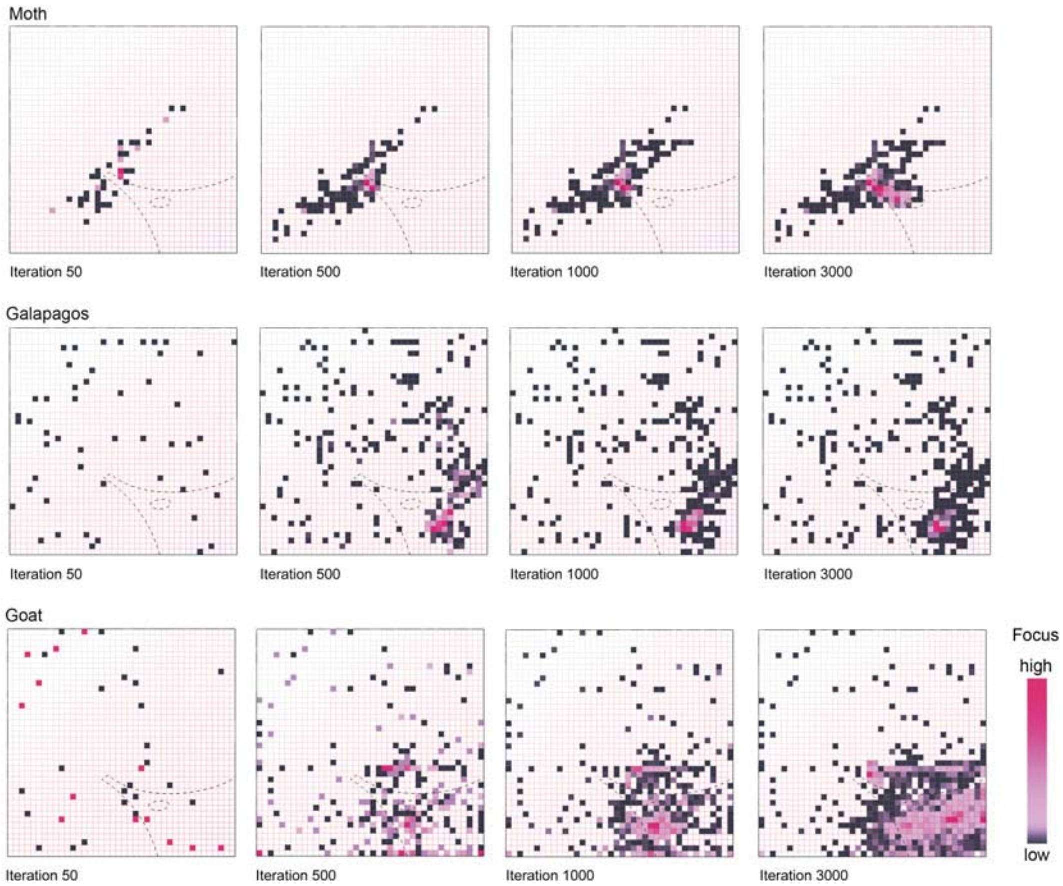


Fig. 6 Search space *C*. *Top* Moth, *middle* Galapagos, *bottom* Goat

agents will seek to disrupt one another's attempts to optimise their own criterion, but over time a multivariate optimum (a convergence) may be found. This is illustrated in Fig. 7. The control of hyper parameters, changing behavioural attributes of agents during optimisation, is unique to ABMs. Moth allows the building designer to change focus and alter agents' search interests simply by altering the agents' behavioural attributes during optimisation. The designer is allowed to manipulate, remove, add, move, and put weights on individual agent-pairs during the optimisation process. This means that the user participates actively in finding a multidimensional balance of the systems. For example, it is possible to change their speed and amount of attraction towards windows or the user may change the objectives without restarting the optimisation process (which is needed when using Galapagos or Goat). This means that the user is able to modify the limits of the search space during the optimisation. In the test case we did not change the objectives at any point. This can be seen in Fig. 8, where the dashed lines represent the constant objectives: $O_{daylight}$, O_{cost} and O_{energy}.

Moth-agents will be attracted or flock around geometrical features in the model representing the design variables. In our case the variables are the windows and a proxy point placed in the centre of the building representing the window type. The number of active agents assigned to a particular design variable will define the focus of the search. In our case we assigned three agent pairs to each of three design variables. As a consequence, we allow each agent-pair to equally

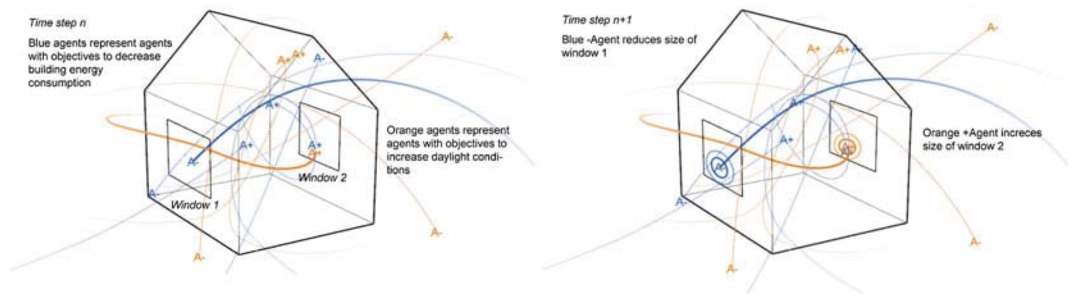


Fig. 7 Example of Moth-agents in action. *Left part* shows time step 1 and *right part* shows time step 2

modify every design variable and we do not distinguish between the objectives.

In Fig. 7 we illustrate how Moth works during optimisation. Here, to make the process simpler, two types of the agent-pairs and two variations of the building geometry are illustrated. The two types of agent-pairs seek to modify the design variables to reach their individual objectives; daylight and energy consumption. The orange Moth-agent objective is to increase daylight by affecting window openings in the building. The Moth-pairs will be attracted towards windows and each +agent will try to increase the window size, while each −agent will do the opposite. Based on the feedback, the +agent will over time get promoted (as larger windows will let more daylight inside the building), and the +agent it will try to further increase the windows sizes.

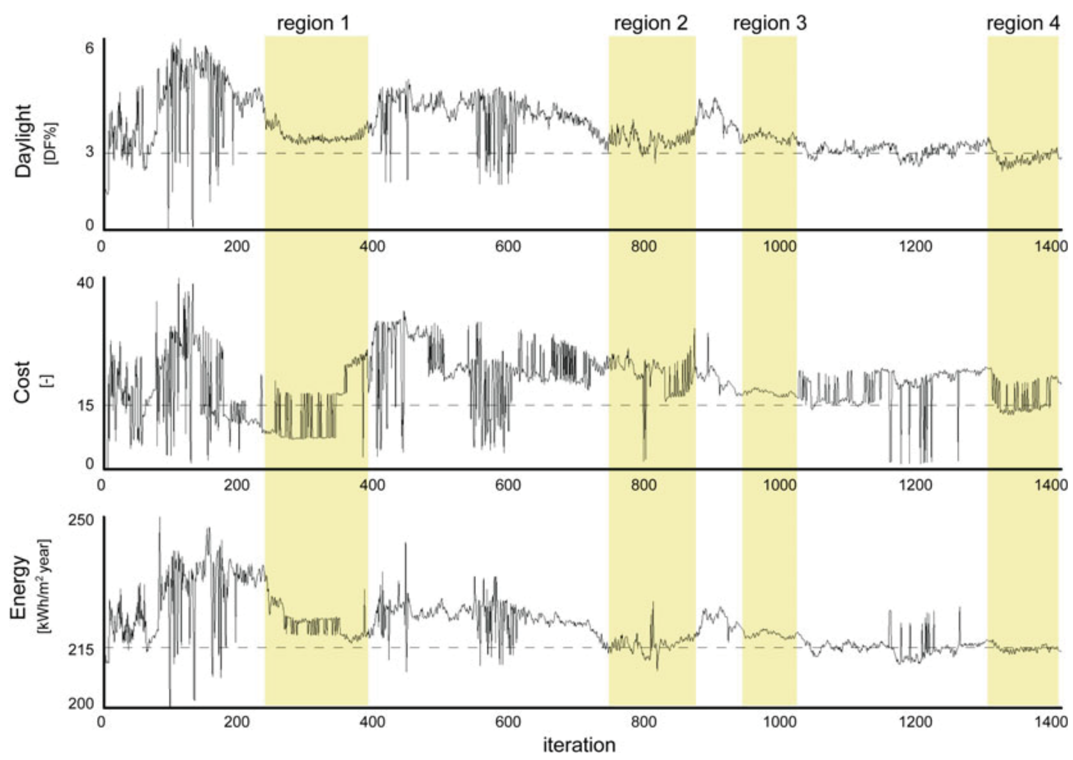


Fig. 8 The multivariate optimisation process using Moth. Regions *1–4* are highlighted to explain certain properties of the Moth algorithm

The −agent will on the other hand have decreasing success in affecting geometry, as it will be gradually demoted during the process. The promotion can be explained as the Moth-agents' 'strength of focus' on the attracting geometry—here windows. The +agent will have more success in circling and thus hitting and affecting the windows that provide most daylight by changing (increasing) the size, whereas the −agent will seem to seek other areas to improve its own goals, that is reducing window sizes, with little success. Over time the +agent will increase windows that make greatest impact on the daylight evaluation, this process is considered a *single objective optimisation*. However, in our case we have three types of agents with three objectives as seen in Fig. 8.

In Fig. 8 four regions are highlighted. The highlighted regions show situations where optima or near optima have been found. What is interesting to notice, is the way Moth concentrates its search in areas where all three objective conditions are close to be fulfilled. Only region 3 maintains a gradient-like search, while the three other regions are "victims" of discontinuity approximations from the cost functions. This is mainly caused by the cost function, $f_{cost}(x)$, of window costs, which are implemented as a jump discontinuity by its enumerable design variable controlling the window types. Region 1, 2 and 4 have at least one objective that oscillates over its defined objective, here jumping back and forth between two window types with different properties. The convergence is never fully adapted to all objectives, but it is seen from region 2 and 4 that convergence of the daylight and energy objectives shares the same gradients while the cost of windows oscillates quite dramatically.

Discussion and Further Research

Seen from the comparison of Moth with two other optimisation algorithms accessible for designers in the early design stage, Moth is a solid choice in terms of accuracy, speed and ability to find multiple optima and near optima in the search space. However, more work is needed to make Moth efficient in finding small minima in larger search spaces. In our multivariate case study in Part 2 we did not find a simultaneous convergence between all three objectives; in other words, we did not find a multivariate minimum, where the three objectives are equally important. However, as the search space is unknown, we do not know if there are any of these minima, and furthermore the demonstration showed that Moth is able to balance two objectives near the optimum while attempting to adapt a third in the process.

From these qualitative tests, the Moth-algorithm seems very efficient in homing in on gradients. There are several explanations why this is happening. First, as the user defines one or several objectives, the algorithm has a measure of "how far" its current performance of a particular solution is from the objective. This is directly translated to the behaviour of the agents, thus making agents far from objectives "more active" and those close to their objective "less active" (and thus less willing to change the design variables). Another explanation is much more problematic, as it applies to the method used to qualify the algorithm. As Moth is predominately applying its search in the "diagonal space" of the search space it is far more likely to find minima in these areas, than outside. The reason for this phenomenon is found in the current implementation of the algorithm. The way "promotions" are distributed from agent-pairs to the design variable vectors are uniformly distributed. There may be any number of agent-pairs and any number of design variables with any number of vector states. The current implementation does not differentiate between the design variables by other means than a relative impact from evaluation to evaluation, which is translated into a relative change in vector states. This creates a parallel shift in all design variable vectors by a factor in a negative or positive direction, which strengthens the diagonal distribution of searches in a two dimensional search space. All three search spaces in part 1 have at least one cluster of minima within the "line of sight" of the algorithm. Therefore it is likely that the performance of Moth is overestimated. Consequently, there are reasons to evaluate ABM-methods like Moth in a more quantitative way. Specifically the comparison of AMBs such as Moth to other methods like Pareto post-balancing methods suggested by e.g. Wang and Zmeureanu (2005) may be necessary to really evaluate Moth as a viable multi objective optimisation method.

A second concern of the ABM method is the oscillation effects. If the agents approach the gradient from a specific angle the search space algorithm can easily get stuck in a multi-dimensional recoil effect. In two dimensions this would look like two very unimaginative or equally skillfull players playing back and forth on a tennis court where they never miss a hand. The reason for this effect is the simplistic nature of the promotion system. Once an evaluation of an agent-pair goes from positive to negative, there is a small chance that the next evaluation again will flip the negative to a positive agent again.

The Moth-agents find only one in infinitely many balances when the optimisation is stopped. But more important to notice is the one solution, is a solution the designer has "moulded" into his or her own personal preference. It can be argued, that this "moulding process" or hyper control of agents is in fact part of the optimisation process, where the qualitative (un-quantifiable) objectives have become an integrated part of the multivariate design domain. We argue that the main reason why ABMs are more relevant for the early design stage, opposed to optimisation methods that depend on post processing, is the direct control of design variable boundaries and control of hyper parameter space during optimisation. ABMs such as Moth allow: (1) the designer to re-evaluate own objectives during the optimisation procedure, (2) the designer to change search focus to dedicated areas of the search space by manipulating behavioural attributes, (3) the optimisation procedure to be part of an exploratory process.

Conclusion

It is concluded that agents may provide valuable design feedback on building performance (e.g. energy, capital cost and indoor environment). Agents may help optimise open ended multivariate design problems, however, the presented system is not particularly efficient in doing so. Given the many fixed and dynamic constraints as well as discontinuities in design variables the algorithm has difficulties in finding true global multivariate optima. However, the presented agent-based system is fully open and adaptable and will allow a high degree of operator intervention during optimisation. For this reason these agent-based optimisation algorithms such as Moth is found better suited in the decision support of early design stages in contrast to traditional stochastic optimisation algorithms e.g. genetic optimisation, which does not support operator intervention during optimisation.

Acknowledgments The work is part of PhD research currently held at the Technical University of Denmark, and is performed in collaboration with Grontmij and Innovationsfonden.

References

Barbati M, Bruno G, Genovese A (2012) Applications of agent-based models for optimization problems: a literature review. Expert Syst Appl 39(5):6020–6028. doi:10.1016/j.eswa.2011.12.015

Bento J, Feijó B (1997) An agent-based paradigm for building intelligent CAD systems. Artif Intell Eng 11 (3):231–244. doi:10.1016/S0954-1810(96)00045-3

Caldas L, Norford L (2001) Architectural constraints in a generative design system: interpreting energy consumption levels. In: Seventh international IBPSA conference, pp 13–15

Danish Building Regulations (2013) Lavenergiramme for kontorer, skoler, institution m.m. Section 7.2.4.2. http://bygningsreglementet.dk/br10_03_id161/0/42

Davidsson P, Persson JA, Holmgren J (2007) On the integration of agent-based and mathematical optimization techniques. In: Proceedings of the 1st KES international symposium on agent and multi-agent systems: technologies and applications, pp 1–10. doi:10.1007/978-3-540-72830-6_1

Gaspar-Cunha A, Loyens D, van Hattum F (2011) Aesthetic design using multi-objective evolutionary algorithms. In: Evolutionary multi-criterion optimization, Springer, pp 374–388

Kaelo P, Ali MM (2006) Some variants of the controlled random search algorithm for global optimization. J Optim Theory Appl 130(2):253–264

Lin S, Gerber DJ (2014) Evolutionary energy performance feedback for design: multidisciplinary design optimization and performance boundaries for design decision support. Energy Build 84:426–441. doi:10.1016/j.enbuild.2014.08.034

Liu B, Wang L, Jin Y-H, Tang F, Huang D-X (2005) Improved particle swarm optimization combined with chaos. Chaos Solitons Fractals 25(5):1261–1271. doi:10.1016/j.chaos.2004.11.095

Negendahl K (2014) Termite. http://cobalab.dk/2014/07/15/termite-1-0/. Retrieved from http://cobalab.dk/2014/07/15/termite-1-0/

Negendahl K (2015) Building performance simulation in the early design stage: an introduction to integrated dynamic models. Autom Constr 54:39–53. doi:10.1016/j.autcon.2015.03.002

Price WL (1983) Global optimization by controlled random search. J Optim Theory Appl 40(3):333–348. doi:10.1007/BF00933504

Robert McNeel & Associates (2013) Rhino. Robert McNeel & Associates. Retrieved from http://www.rhino3d.com

Roudsari MS, Pak M, Smith A, Gill G (2013) Ladybug: a parametric environmental plugin for Grasshopper to help designers to create an environmentally-conscious design. In: 13th Conference of international building performance simulation association

Rutten D (2010) Evolutionary principles applied to problem solving using Galapagos. In: AAG10, Vienna

Salminen M, Palonen M, Sirén K (2012) Combined energy simulation and multi-criteria optimisation of a LEED-certified building. In: Building simulation and optimization conference. Loughborough, UK, pp 372–377

SBI (2013) Be10. SBI. Retrieved from http://www.be10.sbi.dk/

Simon F, Schmiedhofer H, Reis M (2015) Goat. Retrieved from http://www.rechenraum.com/en/references/goat.html

Viehland D, Hardy J, Deville J, Schementi J, Rome K, de Icaza M, Matousek T (2015) IronPython the Python programming language for the .NET Framework. Retrieved 30 April 2015, from http://ironpython.net/

Wang W, Zmeureanu R (2005) Two-phase application of multi-objective genetic algorithms in green building design. In: Ninth international IBPAS conference, pp 1323–1330

Wetter M (2011) A view on future building system modeling and simulation. In: Building performance simulation for design and operation, (i). pp 1–28. doi:10.4324/9780203891612

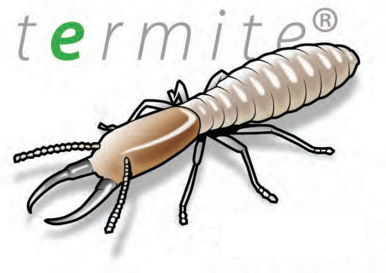
Wetter M, Polak E (2004) A convergent optimization method using pattern search algorithms with adaptive precision simulation. In: Building services engineering research and technology, vol 25(4), p 327. Retrieved from http://bse.sagepub.com/content/25/4/327.abstract

Parametric City Scale Energy Modeling
Perspectives on using Termite in city scaled models

CITIES – Centre for IT Intelligent Energy **Systems in Cities**

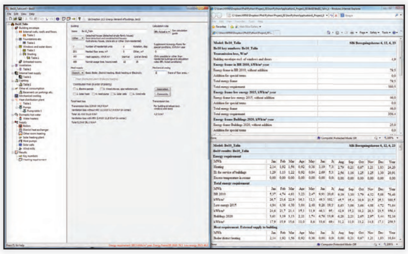
DTU

Termite is a parametric tool using the Danish building performance simulation engine Be10 written for the Grasshopper3D/Rhino3D environment. The tool Be10 is originally intended for building energy frame calculations and is required by Danish law when constructing new buildings. Termite opens up for fully parametric district- and city-size simulations of yearly building energy consumption with the same precisions of energy use as the tool simulates on each and every building. The poster demonstrates some of the parametric flexibilities in using Termite e.g. planning for optimal synergetic envelope requirements, placing solar energy production facilities etc.

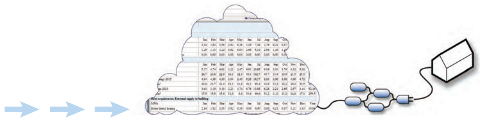


TERMITE MAKES PEOPLE WORK TOGETHER TO DESIGN BETTER BUILDINGS

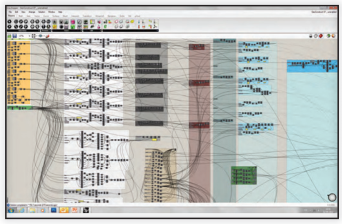
Termite is able to simulate the dynamics of building energy consumption over the year, which incluedes thermal transport, natural and mechanical ventilation, cooling and heating systems, heatpumps, solar cells and much more.



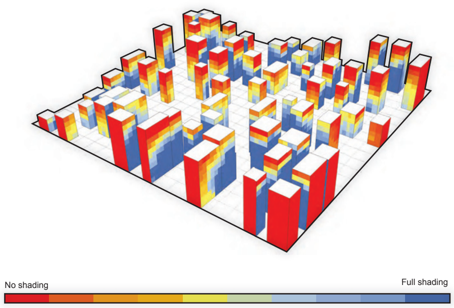
Be10 is a mono-zone, monthly based building performance simulation tool based on the Danish national energy standards of BR10



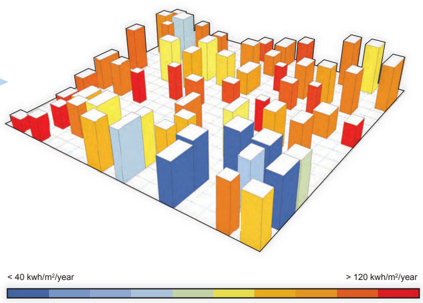
Termite uses the calculation engine of Be10 that accepts every input as a parametric varible in the programmable scripting environment of Grasshopper. Basically all geometric relationships can be modelled and visualized in the CAD environment of Rhino, while user defined algorithms can handle any system setting and input in Be10. While the tool is originally intended to be used in calculating the energy consumption of single buildings, Termite open up for multiple building simulations, thus providing a very efficient tool for city scale energy analysis.



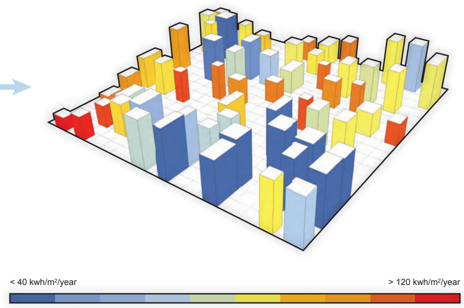
Termite is built by Ph.D candidate Kristoffer Negendahl in collaboration with Grontmij Architectural Engineering Denmark in the quest for qualifying sustainable buildings, districts and cities in the early stages of the design process.



This illustration is showing the self shading mechanisms in a fairly condensed city scape. The shading is calculated with an isovist-method based on C. Reinhart insolation distribution used in Radiance. The shadow effects are then used by Termite in calculating the monthly heat gain through window openings.



Termite can be used to calculate the monthly and yearly energy consumption in kWh/m², here displayed as a colour of building energy consumption. The entire city site is simulated within 5 seconds on an ordinary desktop machine, thus making the tool ideal for parametric design exploration purposes.



Fundamental changes in heating strategies and ventilation requirements can lead to very different energy consumption. Here is the district heating exchanger efficiency improved by 10% and criteria on window g-values is changed in all buildings.

Termite provides very detailed toolsets to model and analyze large scale building energy problems. Effects from building-to-building relationships can be defined by custom user-algorithms with Grasshopper while utilizing Termite to provide valid feedback of each building energy production and consumption. Energy reductions/increases can easily be visualized and Termite may be used as an effective tool for energy planning purposes.

Ph.D student Kristoffer Negendahl
Supervisor Toke Rammer Nielsen

Contact: krnj@byg.dtu.dk | +45 2670 4550

Section for Building Energy
Department of Civil Engineering

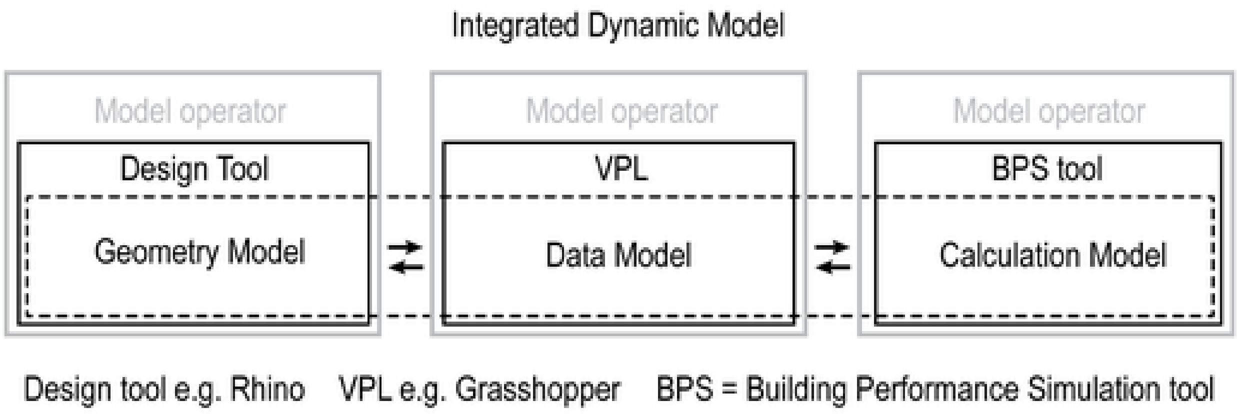
termite® | Architectural Engineering | Grontmij

CITIES
Centre for IT Intelligent Energy Systems

Building Performance Simulation with Grasshopper

Many designers, architects and engineers have been using Grasshopper as a part the building design proces.

This survey explores how building performance simulations are used to design buildings. More specifically, how you use and want to use visual programming languages like Grasshopper with building performance simulations in the early design stages.



The following questions address the use of **integrated dynamic models**, which are defined as:

"An integrated dynamic model is basically a combined model composed of a geometric model controlled in a design tool dynamically coupled to a visual programming language (VPL), which is again dynamically coupled to a building performance simulation (BPS) environment." [1]

Where the Design tool, the VPL and the BPS are:

- **Design tool** : e.g. Rhino
- **VPL** : e.g. Grasshopper
- **BPS** : e.g. Robot, Energy+ or Radiance...*

*often accessed through plugins like Ladybug, Geco...

Name

Your name will only be used to identify duplicate posts.

Company/University

Company/University name will only be used to identify business/research area

Dicipline Select current discipline of work ▾

Click button to start the survey:

Integrated dynamic models used in research and practice ▽

Integrated dynamic models used in research and practice

Please think of the latest project where you have used (or participated in using) intergrated dynamic models.

Which of the following arguments for that particular project are true?

Project type

- ○ Commercial project
- ○ Academic/research project
- ○ Other

Building type

- ○ Office building
- ○ Educational building
- ○ Residential building
- ○ Mixed-use building
- ○ Other

Building size

- ○ 1-250 m2
- ○ 250-500 m2
- ○ 500-2000 m2
- ○ 2000-5000 m2
- ○ 5000+ m2

Have the interactions between members of the design team affected by the use of integrated dynamic models in this project? If so, how?

0/250

Has the use of integrated dynamic models had an impact on the simulated performance such as energy consumption, indoor environment and/or sustainability of this project? If so, how?

0/250

Has the use of integrated dynamic models had an impact on other decisions in this project? If so, which?

0/250

To what extent for this particular project do you agree on these five statements A), B), C), D) and E)?

Statement A) The use of integrated dynamic models have created a better common starting point of collaboration between the members in the design team

Select and drag the sliders. Where [0] is "I do not agree", and [100] is "I agree completely".

Statement A) 50

Statement B) The use of integrated dynamic models have improved the communication between the members in the design team

Select and drag the sliders. Where [0] is "I do not agree", and [100] is "I agree completely".

Statement B) 50

Statement C) The use of integrated dynamic models have improved the (simulated) performance of the building design

Select and drag the sliders. Where [0] is "I do not agree", and [100] is "I agree completely".

Statement C) 50

Statement D) The use of integrated dynamic models have assisted in the exploration of architectural expressions and concepts of the building design that is not directly associated with the (simulated) building performance

Select and drag the sliders. Where [0] is "I do not agree", and [100] is "I agree completely".

Statement D) 50

Statement E) The use of integrated dynamic models have positively assisted in achieving high performing buildings without compromising the architectural expressions and concepts of the building design

Select and drag the sliders. Where [0] is "I do not agree", and [100] is "I agree completely".

Statement E) 50

Click button to continue the survey:

General use of integrated dynamic models

How do you ***use*** integrated dynamic models as a tool in the early design stage? As a:

Select and drag the sliders. Where [0] is "I do not agree", and [100] is "I agree completely".

Sketching tool	50
Communication tool	50
Calculation tool	50
Collaboration tool	50

How would you ***like*** integrated dynamic models ***to be used*** as a tool in the early design stage?: As a:

Select and drag the sliders. Where [0] is "I do not agree", and [100] is "I agree completely".

Sketching tool	50
Communication tool	50
Calculation tool	50
Collaboration tool	50

When using integrated dynamic models in the early design stage, what is more important:

Select, drag and arrange the order of the characteristics of the model, where [1] is more important and [5] is least important:

1: Flexibility (of features and complexity)

2: Precision (of the BPS)

3: Speed (of performance feedback)

4: Usability (i.e. easy to use)

5: Visual quality (for presentation)

Specifically concerning BPS requirements, what is more important:

Select and click one button, if more important. Click both buttons if equally important.

Speed	Precision

Specifically concerning visual feedback requirements, what is more important:

Select and click one button, if more important. Click both buttons if equally important.

Speed	Visual quality

Specifically concerning VPL requirements, what is more important:

Select and click one button, if more important. Click both buttons if equally important.

Flexibility	Usability

If you have any notes or comments to the researchers, please feel free to put them here:

Click button to submit the survey:

Submit

Discipline
A – Architect
E – Engineer
O – Other

Grasshopper / Dynamo Forums 15 respondents

Have the interactions between members of the design team affected by the use of integrated dynamic models in this project? If so, how?

A	Yes. A significant focus for the research was creating a protocol for facilitating the exchange of information across scales and instances within a networked model.
A	not relevant, I was the only team member
E	no, was a university workshop and we divided the things to do
E	not relevant, I've used it for experimentation.
A	Yes. Became more aware of the energy impact of their design
O	not relevant. Product manufacturer, I work to create new solutions.
O	Yes. Indeed. Having analytical models downstream from geometric/generative models becomes a shared way in which we discuss design options (in terms of what makes a good fit candidate in relation to somewhat
A	not relevant Reseach project. Just me working!
A	We are using Dynamo for Revit to extract information from models in an effort to "automate" repetitive calculation tasks for developer proformas and code requirements. This frees up more time for thoughtful design decisions.
E	Independecies between models makes sure that things has to be coordinated.
E	Improved ability to consider options and make design decisions as a wider team
A	Yes - It was a big help for the discussion through out the design process (models were applied after concept stage)

Has the use of integrated dynamic models had an impact on the simulated performance such as energy consumption, indoor environment and/or sustainability of this project? If so, how?

A	Yes. But not so much sustainability, but very much so for structural performance analysis and tectonic expression
A	No. As it was just sketch without simulated performance visualization. I was interested in result shape which was the result of partial environmental simulation and totally satisfied me.
E	Yes. Of course. It reduced the use of energy but this was not the only way that we used for improve the performance so I don't know how much it affected the project.
E	Yes, it has. Developing the experiment I had an improvement of the initial condition (energetically speaking), driving my architectural choices till environmental sustainable task.
A	Yes. Adaptation of the design of facade panels based on the aquired analysis
D	Yes. The fine tuning of daylight systems was done there.
A	Yes. The analysis allowed an optimization of the building up to 1/4 better energy consumption values
O	Yes. Dynamic passive strategies always have an impact.
O	Yes. In particular in relation to structural analysis (purely geometric analysis as well as FEA). When working with form active structures this becomes especially crucial and was arguably the primary design driver.
A	Yes. the use of these models helps in real-time feedback to improve the design proposal. this has proved very useful especially when used with Galapagos/Octopus for evolutionary algorithms to find best results among conflicting design criteria
A	I have not tried this yet.
E	Yes, ability to consider great number of proposals.

A	We achieve a higher glass procentage than expected in the concept stage

Has the use of integrated dynamic models had an impact on other decisions in this project? If so, which?

A	Yes...they worked for us during the fabrication and construction phases, which (as a research project exploring specific material behaviours) was executed in house.
A	No. As it was just sketch without simulated performance visualization. I was interested in result shape which was the result of partial environmental simulation and totally satisfied me
E	Yes. However, the impact is the model. All our study was based on the integrated dynamic part so, it not had any impact, it was the main purpose of all ours works.
E	No.
D	Yes - it had an impact on the calculation of materials and an entry portal
A	After the analysis the building structure had to be revised
O	Yes. Mostly through visualization of the results.
O	Yes. Depending on the definition of dynamic model, the form finding process could be considered a simulation model (and is certainly dynamic in both the traditional computer science and pure science way). This was
A	Yes. Sometimes the limitations of the tools forces the designer to change the model accordingly simply because it cannot simulate this or that feature
A	The use of real-time analysis via Dynamo has created a sense of accountability for what is being modeled. As opposed to the previous post-analysis method, we now implement more efficient strategies and can see the immediate results of decisions.
E	Cost benefit of various design options
A	Yes economy and architectural quality (e.g. daylight). The tools took us into an early discussion about the price of the facade / building parts vs. indoor climate and energy consumption vs. the architectural quality (the developers product)

Nordfløjen	**6 respondents**

Have the interactions between members of the design team affected by the use of integrated dynamic models in this project? If so, how?

E	There was a need for a common understanding of why we made the model, an what the benefits would be. Therefor there was a much greater understanding and respect for the different architectural and engineering disciplins.
E	Arkitekten kunne måske bedre se konsekvensen/muligheden for designet når parametre blev ændret og justeret.
A	Yes, facadedesign.
E	Yes, by optimising both process and outcome in therms of daylight utilisation and energy performance
A	Yes very much, particularly discussions regarding design cosequences on the architects part versus daylight and energy performance on the engineers part. Also how to integrate and sharing data between our different models.
D	Yes, btw. architects and engineers. DM provided a communal "language" that "translated" engineers analysis into tangible design impact. The speed of calcs implementation and accelerated design choices.

Has the use of integrated dynamic models had an impact on the simulated performance such as energy consumption, indoor environment and/or sustainability of this project? If so, how?

E	The daylight is optimized in every room, because the model made it possible to vary the windowsizes more when taking into consideration the amount of sun and shadow on every part of the facade and because of the detailed location of the roomfunctions
E	Svært af svare på da det faktiske energiforbrug aldrig er blevet beregnet og da der aldrig er lavet en bæredygtighedsanalyse. I designfasen var det nok nemmer at håndtere den komplekse geometri til Be10 beregningen.
A	Cannot answer to that question.
E	Yes, by optimising both process and outcome in therms of daylight utilisation and energy performance
A	Yes, we couldn't have done it without parametric tools due to the complexity of the design. Window sizes and distributions was a direct result of the energy calculations.
D	I believe it has become more precise. Indirectly it might have increased the performance due to the increased level of implementation in the design.

Has the use of integrated dynamic models had an impact on other decisions in this project? If so, which?

E	Placering af solafskærmning.
A	No
E	It made it possible to work with optimised models and solutions, especially in the early stage. That is a clear improvement in relation to 'standard' or old-fasioned designprocess, wherer we'll rely on 'rules of tumb' and general experience only
A	Not really, but I think it reinforced the idea of using a set of standard facade elements distributed according to energy/daylight performance, in a seemingly random fascion.
D	Yes, it has made it easier and faster to calculate the construction costs and total budget.

Extra comment	I'll stress the possibility of working with different disiplines simulstaniously, ligh daylighting, indoor climate, energy performance ect. AND architectural form and expression (functions and facades).

Batteriet	**3 respondents**

Have the interactions between members of the design team affected by the use of integrated dynamic models in this project? If so, how?

E	Arbejdede lang tætter sammen end på noget projekt tideligere
E	Not enough involved.
A	Yes, particularly discussions regarding design cosequences on the architects part versus energy performance on the engineers part. Also how to integrate our different models.

Has the use of integrated dynamic models had an impact on the simulated performance such as energy consumption, indoor environment and/or sustainability of this project? If so, how?

E	Ja, det gjorer det muligt, at løse projekter - som dog aldrig blev udført.
E	Not enough involved.
A	Yes, window sizes and locations were a direct result of the energy calculations, eventhough at this point very sketchy and imprecise.

Has the use of integrated dynamic models had an impact on other decisions in this project? If so, which?

E	Hele facade designet
E	Not enough involved.
A	Not that I know.

Good Year **4 respondents**

Have the interactions between members of the design team affected by the use of integrated dynamic models in this project? If so, how?

O	Yes - en del medarbejdere på holdet fik åbnet øjnene for betydningen af mindre ændringer i forudsætninger for facade design kan have stor betydning for energiberegninger og anlægsomkostninger
A	Don't know. I was very little involved in the integrated dynamic modelling.
D	Inspired the architects to take interest in integrating the performance-based demands. Increased the early-phase performance guidelines. Accelerated the cost calculations and construction budgets.

Has the use of integrated dynamic models had an impact on the simulated performance such as energy consumption, indoor environment and/or sustainability of this project? If so, how?

O	Teoretisk ja, men det er svært at afgøre, da projektet kun var et konkurrence projekt
E	fast rsponse on alternatives
A	I was very little involved in the integrated dynamic modelling. But I think we couldn't have done the energy calculations without parametric tools due to the complexity of the design.
D	Don´t remember

Has the use of integrated dynamic models had an impact on other decisions in this project? If so, which?

O	ja - det blev brugt aktivt i de første faser, men arkitekten ændrede i sidste fase, så en del af de optimerede vinkler forsvandt.
E	Facade
A	Don't know. I was very little involved in the integrated dynamic modelling.
D	Cost and construction calculations. Increased the design options surprisingly, eg. enabled a design with an expansive glass facade

Nordic Built	**3 respondents**

Have the interactions between members of the design team affected by the use of integrated dynamic models in this project? If so, how?

A	Don't know. I was very little involved in the integrated dynamic modelling.
D	Yes, it has inspired architects to work creatively with performancebased demands from pre-design stage. It directed the programming of functions. It helped to implement and communicate a complex sustainability concept.

Has the use of integrated dynamic models had an impact on the simulated performance such as energy consumption, indoor environment and/or sustainability of this project? If so, how?

A	Don't know. I was very little involved in the integrated dynamic modelling.
D	Don't remember

Has the use of integrated dynamic models had an impact on other decisions in this project? If so, which?

A	Don't know. I was very little involved in the integrated dynamic modelling.
D	Yes, it threw a diverse perspective on the indoor climate concept, which enhanced the reflections of the design team in terms of directions. It generated a complex sustainability strategy and communication tool.

Kirk **1 respondent**

Have the interactions between members of the design team affected by the use of integrated dynamic models in this project? If so, how?

A Not really. This was to test an integrated dynamic model on a design that was more or less fixed allready. So the scope was the modelling, not the project itself.

Has the use of integrated dynamic models had an impact on the simulated performance such as energy consumption, indoor environment and/or sustainability of this project? If so, how?

A Yes, very much. A more precise calculation of the energy consumption was possible.

Has the use of integrated dynamic models had an impact on other decisions in this project? If so, which?

A Not that I know of.

Consequence based design is a framework intended for the early design stage. It involves interdisciplinary expertise that secures validity and quality assurance with a simulationist while sustaining autonomous control of building design with the building designer. Consequence based design is defined by the specific use of integrated dynamic models. These models include the parametric capabilities of a visual programming tool, the building analyses features of a building performance simulation tool and the modelling and visualisation features of a design tool. The framework is established to enhance awareness of building performance in the early stages of building design, in the aim to create High-Performance Buildings.

DTU Civil engineering
Department of Civil Engineering
Technical University of Denmark

Brovej, Building 118
2800 Kgs. Lyngby
Telephone 45 25 17 00

www.byg.dtu.dk

ISBN: 978-87-7877-419-4
ISSN: 1601-2907