Technical University of Denmark

DTU

Building energy optimization in the early design stages: A simplified method

Negendahl, Kristoffer; Nielsen, Toke Rammer

Link back to DTU Orbit

DTU Library
Technical Information Center of Denmark

Accepted Manuscript

Title: Building energy optimization in the early design stages:
A simplified method

Author: Kristoffer Negendahl Toke Rammer Nielsen

Please cite this article as: K. Negendahl, T.R. Nielsen, Building energy optimization in the early design stages: a simplified method, *Energy and Buildings* (2015), http://dx.doi.org/10.1016/j.enbuild.2015.06.087

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

Highlights

- Optimization (building energy related) in the early design stage is reviewed
- Quasi-steady-state methods for building energy design are reviewed
- New hourly Quasi-steady-state (HQSS) method is described
- Based on a real problem HQSS is demonstrated for multivariate optimization (with other BPS tools)
- HQSS in integrated dynamic models provide high speed and flexibility needed in the early design stage

ACCEPTED MANUSCRIPT

Main author email: kristoffer.negendahl@gmail.com

Main author telephone number: +45 2670 4550

1

Building energy optimization in the early design stages:
a simplified method

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 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 [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

2

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 & 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 & 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 of 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.

3

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].

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 one hour in the range of a year of 8760 hours. 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 equation (8).

Table 1 BPS tools applied to the method

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;

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

where $Q_{C,nd,cont}$ is the needed amount of cooling to maintain set point temperatures and $\eta_{C,ls}$ a dimensionless utilization factor depended on time constants and used specifically in seasonal and monthly calculation periods [32]. When the maximum cooling capacity, $Q_{C,cap}$ is known, equation (4) can be written as;

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

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

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

where k is the zone and nk is the number of zones in the building, $Q_{equip,k} = 6$ W/m^2 and $Q_{occu,k} = 4$ W/m^2 are 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:

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

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;

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

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

	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

4

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;

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

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 Figure 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, 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;

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

where $A_{win,k}$ is the area and $U_{win,k}$ is the U-value of the window (inclusive frame), $A_{wall,k}$ and $U_{wall,k}$ is 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°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;

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

where $\rho_{air} \cdot c_{air}$ is the heat capacity of air volume set to 1200 J/(m³K), $b_{ve,k}$ is the dimensionless temperature adjustment factor representing the heat recovery rate. $q_{ve,k,max}$ is the maximum airflow expressed in m³/s. The air supply temperature $\theta_{ve,set,i}$ 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 related to cooling outside air to 18°C is not accounted for in the minimization function seen in equation 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

---

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

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.
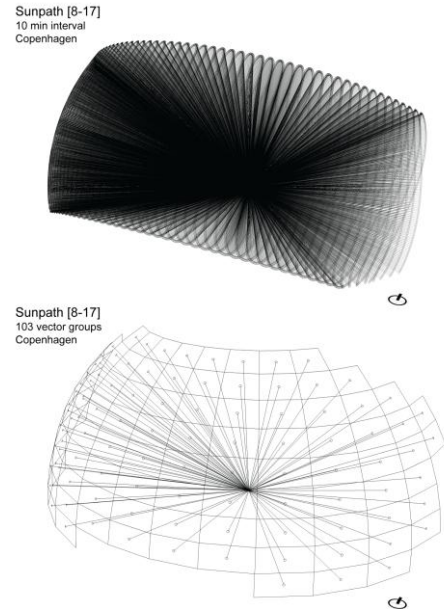

Figure 1 Annual solar sky component generated from the usage profile of a typical office [8-17]. The reduced vector field can be seen in the bottom picture.

---

[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]

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 Figure 2).
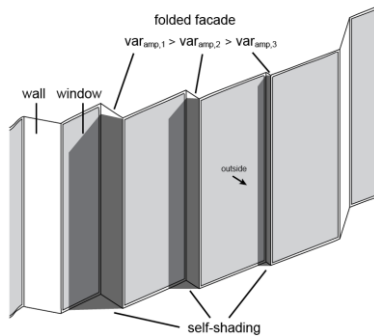

Figure 2 Folded facade concept. The amplitude of the facade folds marked by the variable $var_{amp,1-3}$ create self-shading mechanisms on the neighboring facade unit.

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 Figure 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

---

[3] amount of folding is determined by adjusting amplitude var_{amp} and window size var_{pl} (see Figure 3) and var_{blend} (see Figure 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)

---

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 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 Figure 3 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.
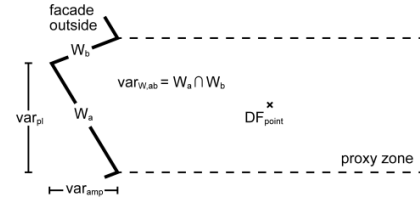

Figure 3 Plan view of a proxy zone represented as a dashed line. The variables are used to constrain the optimization process

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 $var_{amp(1-3)}$, $var_{pl(1-3)}$, var_{blend} to control the folding. Where $var_{amp(1-3)}$ controls the amplitude in on the three facades. $var_{pl(1-3)}$ controls the vertical placement of the fold on each façade and var_{blend} adjust the "blending effect", that intermix the folding between facades. Figure 4 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.
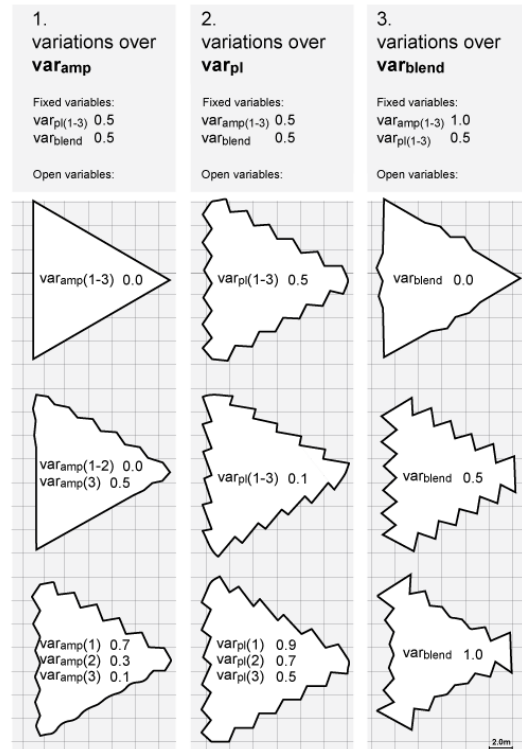

Figure 4 Plan views of a small building example to explain the changes in design variables. 1. shows variations over var_{amp}. 2. shows variations over var_{pl}. 3. shows variations over var_{blend}

6

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 Figure 4 does not represent any architectural preferred strategy, but shows the impact of the design variables.

4.2 Objective functions

Four objective functions $f_{energy}(x)$, $f_{cost}(x)$, $f_{daylight}(x)$, $f_{thermal}(x)$ are minimized by the multivariate optimization algorithm SPEA2 [45,46].

The building energy use; $f_{energy}(x)$ is a function of the annual simulated heating $Q_{heating}$, cooling $Q_{cooling}$, ventilation Q_{vent} and lighting Q_{light} :

$$\min_{x\in\mathbb{R}^s\times\mathbb{Z}^t} f_{energy}(x) = \sum_{i=1}^{ni}\left(Q_{heating,i} + PF\cdot(Q_{cooling,i} + Q_{vent,i} + Q_{light,i})\right) \quad (12)$$

where i is the load condition of the particular condition, ni and is the number of load conditions. $f_{energy}(x)$ 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:

$$\min_{x\in\mathbb{R}^s\times\mathbb{Z}^t} f_{cost}(x) = \left(\sum_{k=1}^{nk}(A_{w,k}\cdot C)\Big/10\right) + 50 \quad (13)$$

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 Figure 6 and 7.

The daylight evaluation $f_{daylight}(x)$ is defined by the CIE uniform sky simulation of a point in the center of the proxy zone, 0.85m from the floor. A penalty function $ERF(DF_w) - 3$ (also shown in Figure 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 Figure 5):

$$\min_{x\in\mathbb{R}^s\times\mathbb{Z}^t} f_{daylight}(x) = nk\Big/\left(\sum_{k=1}^{nk}\left(ERF(DF_{w,k}) - 3\right)\right) \quad (14)$$

where k is the proxy zone and nk is the number of proxy zones in the building. $ERF(DF_w)$ is defined by $ERF(DF_w) = \frac{2}{\sqrt{\pi}}\int_0^{DF_x} e^{-t^2}dt$. DF_x is simulated by Radiance for every solution, x.

The objective function of the thermal requirements is defined as:

In each calculation step t evaluate:
if $\quad (Q_{int} + Q_{sol} - \eta\cdot(Q_{tr} + Q_{ve}) \geq \theta_{C,cap})\quad$ is True
increment overheating hour h $\quad (15)$

$$\min_{x\in\mathbb{R}^s\times\mathbb{Z}^t} f_{thermal}(x) = nk\Big/\left(\sum_{k=1}^{nk}\left(\sum_{t=1}^{nt}(h)\right)\right)$$

where k is the proxy zone and nk is the number of proxy zones in the building, $Q_{int} + Q_{sol} - \eta\cdot(Q_{tr} + Q_{ve}) \geq Q_{C,cap}$ is explained in eq.

(5). $\theta_{C,cap} = \frac{Q_{C,cap}}{A_{proxy}}$ represents the maximum cooling capacity at any hour in the year, set to 40 W/m². A_{proxy} is the area of the proxy zone.
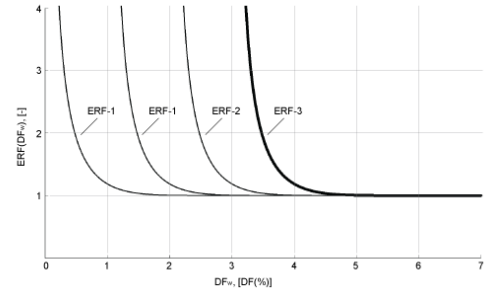

Figure 5 Penalty functions used to limit the influence of very high daylight factors and avoid low daylight factors. Penalty factor, ERF-3 is used in the case study.

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 seconds. This is considered very fast when we are talking whole building simulations on regular PCs.

In Figure 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).
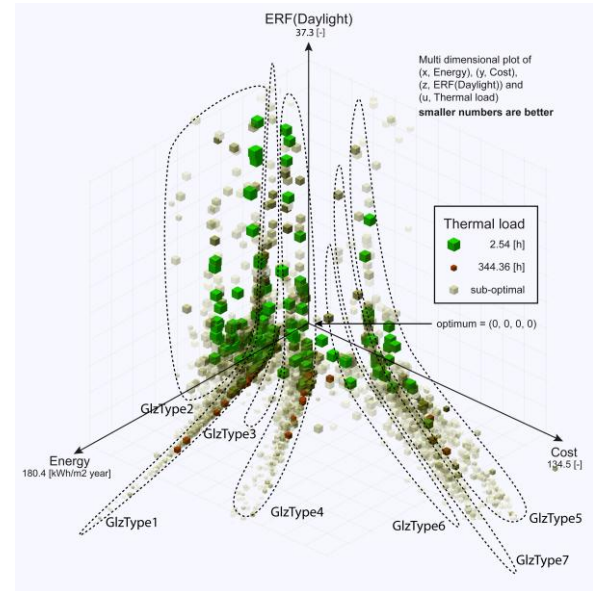

Figure 6 Four dimensional solution space: Energy, Cost, ERF(Daylight) and Thermal load. Sub-optimal solutions are shown in grey colors. Dashed lines encapsulate the solutions associated with individual window-types. Please see Figure 5 for the explanation of ERF and Figure 7 for the explanation of size and color of solutions.
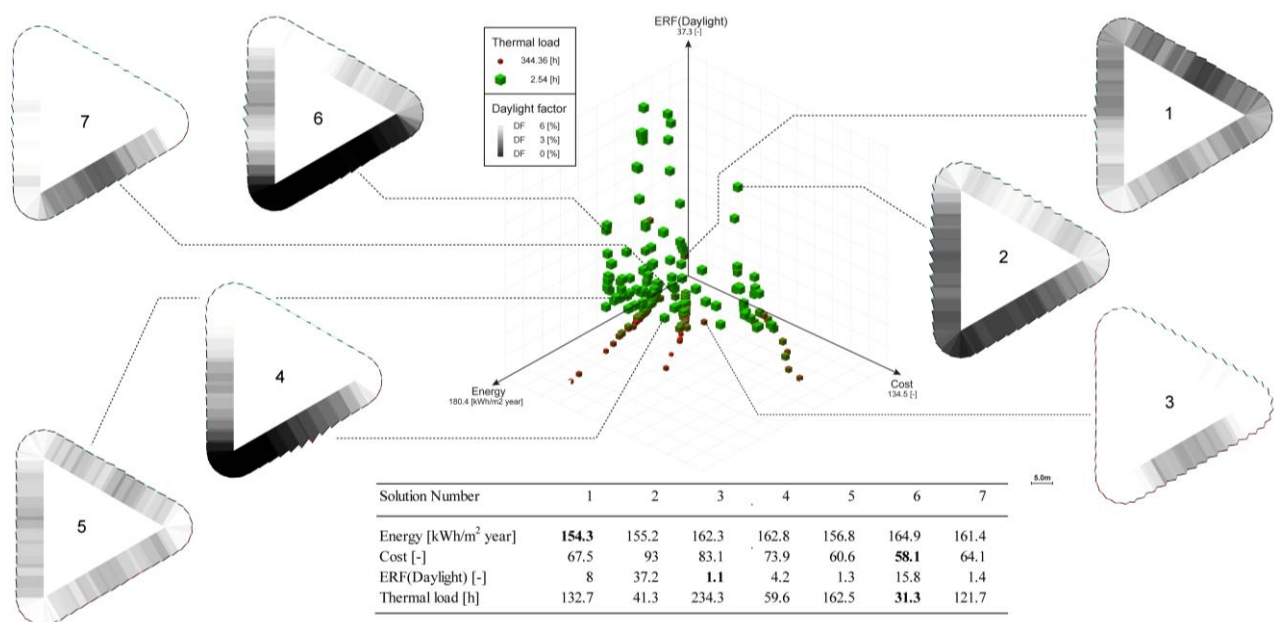
7

Figure 7 The plot in the middle shows axes of a Cartesian space (x, y, z), where x is energy [kWh/m² 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

Solution Number	1	2	3	4	5	6	7
Energy [kWh/m² 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

Each field seen in Figure 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.

Figure 8 Tradeoff between energy and cost plotted with sized points representing thermal loads. 1ˢᵗ generation Pareto front of solutions are shown with a grey line and the 32ⁿᵈ generation Pareto front are shown with a black line

When looking more specifically into the Pareto solutions (see Figure 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 towards north east and close itself towards 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).

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).
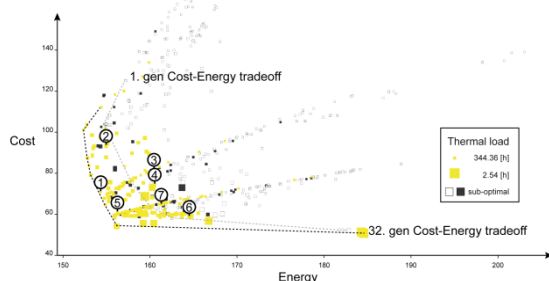
When it comes to cost-benefit analysis of the seven selected solutions the capital cost versus running costs (building energy consumption) is a popular way to choose a particular balanced solution. From Figure 8 the tradeoff between cost (of windows) and cost of annual building energy use is seen. The 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 valued high, but in terms of capital-cost and annual energy costs a uniform and almost flat façade composition is better performing.

Table 2 Window type properties. *Cost index is created for this case study and do not signify real costs.

Name	Configuration	Window type	Uwin (EN10077-1)	Ug (EN673)	G (EN410)	LT (EN410)	Eref (DS418)	Cost* index, C
GlzType1	4-16-4 argon	Double layer window	1.33	1.02	0.52	0.77	-34	1
GlzType2	4-16-4 argon	Double layer window	1.40	1.11	0.62	0.80	-24	1.1
GlzType3	4-16-4 argon	Double layer window	1.53	1.26	0.77	0.82	-11	2.6
GlzType4	4-12-4-12-4 argon	Triple layer window	1.11	0.72	0.50	0.72	-17	1.9
GlzType5	4-12-4-12-4 argon	Triple layer window	0.79	0.5	0.42	0.63	-2	3.6
GlzType6	4-12-4-12-4 argon	Triple layer window	0.82	0.53	0.50	0.72	8	4.7
GlzType7	4-12-4-12-4 argon	Triple layer window	0.90	0.62	0.62	0.73	21	6.1

8

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 seconds 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 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 Figure 1.
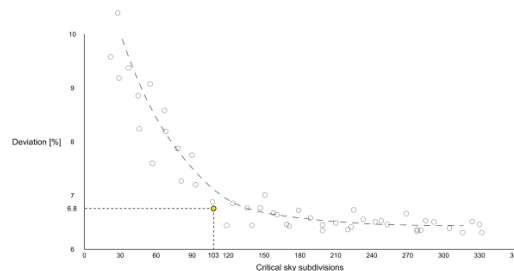
Figure 9 Absolute beam component deviations between Energy+ and HQSS in % when altering the critical sky subdivisions. Sky subdivision used in this article is marked in the plot.

The consequence of altering the solar vector angles is showed in Figure 9 where the number of "critical" sky subdivisions is 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 Figure 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 Figure 10 that HQSS deviations from Energy+ are varying over the orientation and inclination with a bias towards east around the vertical inclination
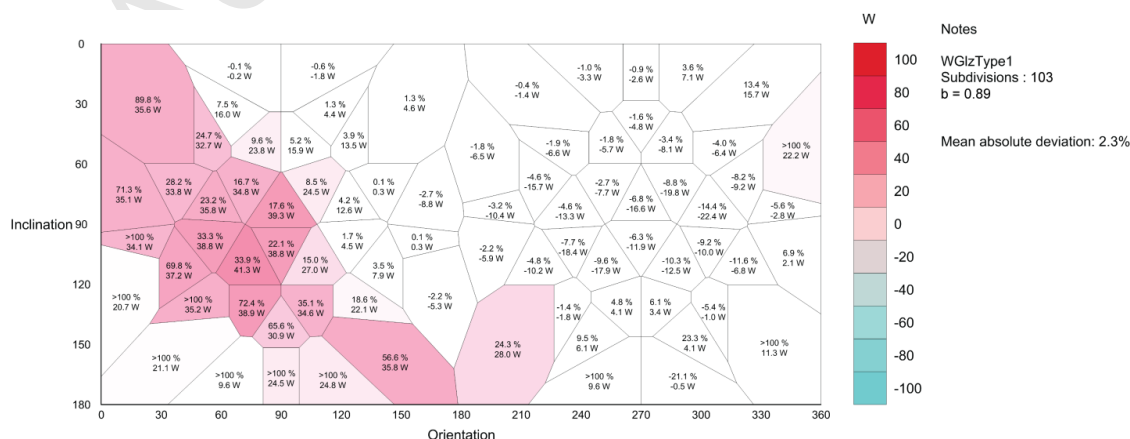
Figure 10 Beam component deviations between Energy+ and QHSS measured in Watts plotted against inclination and orientation of a surface.

9

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

[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.

[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, Proc. 7th Int. Conf. Des. Decis. Support Syst. Arch. Urban Planin. (2004) 2–5.

[5]   A. Hermund, Building Information Modeling in the Architectural Design Phases And Why Compulsory BIM can Provoke Distress Among Architects, eCAADe. (2009) 1–8.

[6]   R. Mora, C. Bédard, H. Rivard, A geometric modelling framework for conceptual structural design from early digital architectural models, Adv. Eng. Informatics. 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. Informatics. 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, Build. Perform. Simul. Des. Oper. (2011) 1–28.

[14]   S. Petersen, Simulation-based support for integrated design of new low-energy office buildings, Technical University of Denmark, 2011.

[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. 1st 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 2014, 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.

10

[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.

[21] W. Wang, R. Zmeureanu, Two-Phase Application of Multi-Objective Genetic Algorithms in Green Building Design, Ninth Int. IBPAS Conf. (2005) 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 (Ed.), 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, (2012) 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, Symp. Simul. Archit. Urban Des. (2010) 65–72.

[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, https://github.com/mostaphaRoudsari/Honeybee. (2014).

[31] K. Negendahl, Termite, http://cobalab.dk/2014/07/15/termite-1-0/. (2014).

[32] ISO, EN ISO 13790:2008 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.

[34] M. Benedikt, To take hold of space: isovists and isovist fields, Environ. Plan. 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. 2012, Springer Vienna, 2012: pp. 67–76.

[41] B. Raphael, Multi-criteria decision making for collaborative design optimization of buildings, Built Environ. Proj. Asset Manag. 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) 1512–1525.

[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. Informatics. 22 (2008) 371–384.

[45] R. Vierlinger, Octopus, http://www.grasshopper3d.com/group/octopus. (2014).

[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, http://bygningsreglementet.dk/br10_03_id161/0/42. 2013 (2013).

[48] Wolfram, ERF, http://mathworld.wolfram.com/Erf.html. (2014).

Inernational Build. Perform. Simul. Assoc., Chambéry, France, 2013: pp. 2357–2364.

11

[49]   C. Souza, I. Knight, Thermal performance simulation from an architectural design viewpoint, Build. Simul. (2007) 87–94.

[50]   K. Negendahl, Parametric design and analysis framework with integrated dynamic models, in: Proc. 3rd 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.

12

	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

Table 2

Name	Configuration	Window type	Uwin (EN10077-1)	Ug (EN673)	G (EN410)	LT (EN410)	Eref (DS418)	Cost* index, C
GlzType1	4-16-4 argon	Double layer window	1.33	1.02	0.52	0.77	-34	1
GlzType2	4-16-4 argon	Double layer window	1.40	1.11	0.62	0.80	-24	1.1
GlzType3	4-16-4 argon	Double layer window	1.53	1.26	0.77	0.82	-11	2.6
GlzType4	4-12-4-12-4 argon	Triple layer window	1.11	0.72	0.50	0.72	-17	1.9
GlzType5	4-12-4-12-4 argon	Triple layer window	0.79	0.5	0.42	0.63	-2	3.6
GlzType6	4-12-4-12-4 argon	Triple layer window	0.82	0.53	0.50	0.72	8	4.7
GlzType7	4-12-4-12-4 argon	Triple layer window	0.90	0.62	0.62	0.73	21	6.1

Figure 1

Sunpath [8-17]
10 min interval
Copenhagen

Sunpath [8-17]
103 vector groups
Copenhagen

folded facade

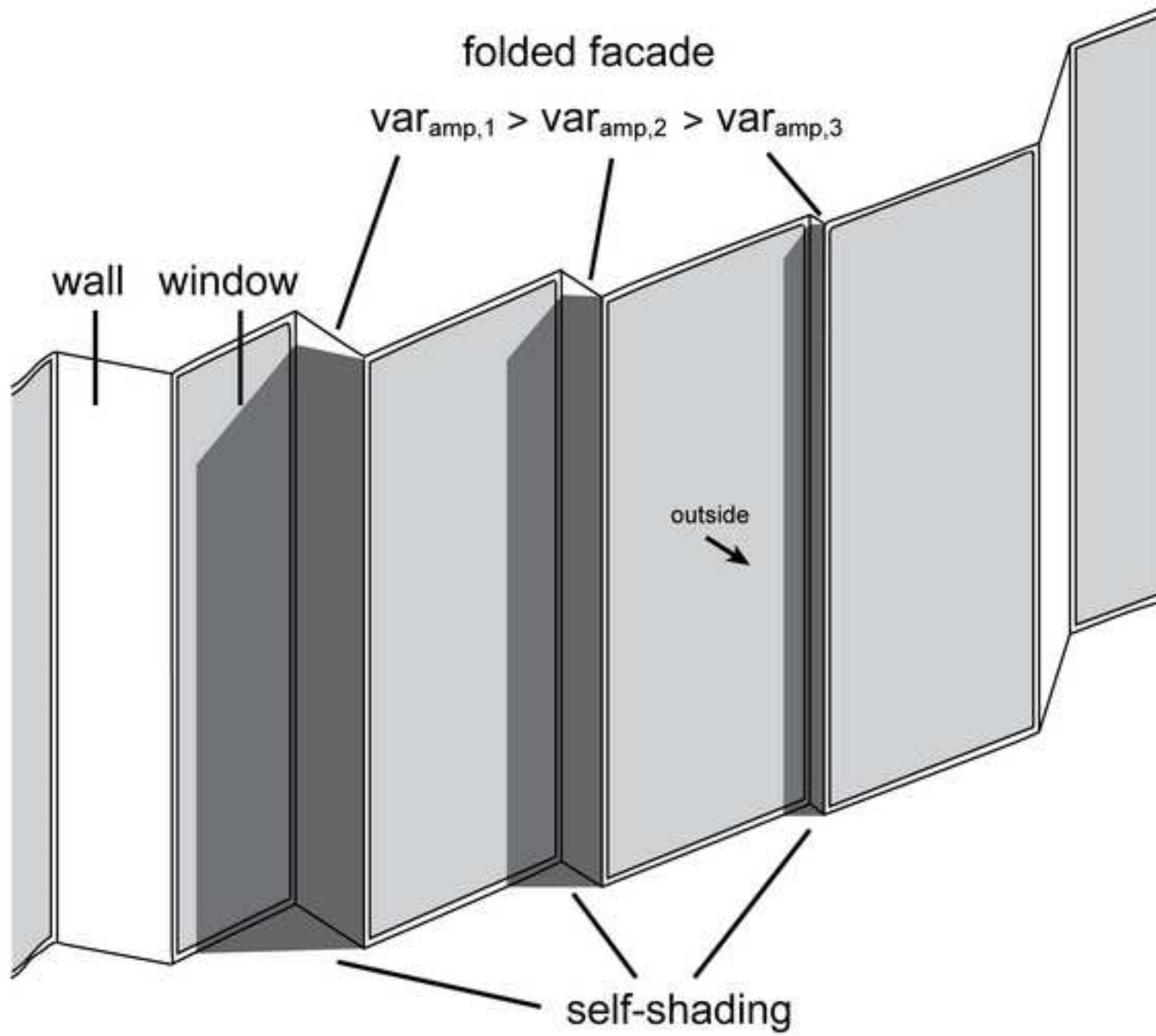$var_{amp,1} > var_{amp,2} > var_{amp,3}$

wall window

outside

self-shading

Figure 3
[Click here to download high resolution image](#)

facade outside

W_b

$var_{W,ab} = W_a \cap W_b$

var_{pl}

W_a

$DF_{point}^{\times}$

proxy zone

$\vdash var_{amp} \dashv$

1.
variations over
varamp

Fixed variables:
$var_{pl(1-3)}$ 0.5
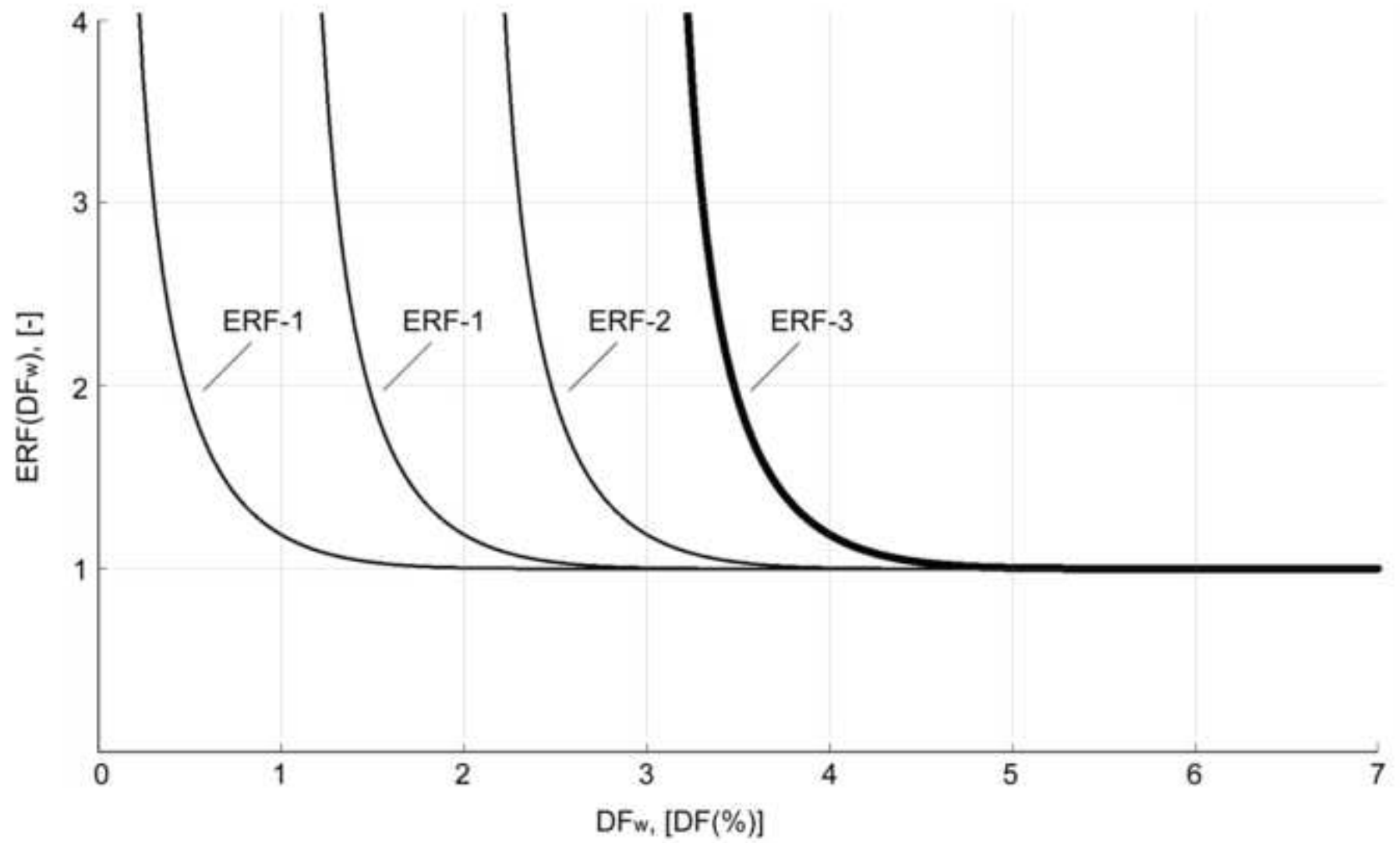var_{blend} 0.5

Open variables:

$var_{amp(1-3)}$ 0.0

$var_{amp(1-2)}$ 0.0
$var_{amp(3)}$ 0.5

$var_{amp(1)}$ 0.7
$var_{amp(2)}$ 0.3
$var_{amp(3)}$ 0.1

2.
variations over
varpl

Fixed variables:
$var_{amp(1-3)}$ 0.5
var_{blend} 0.5

Open variables:

$var_{pl(1-3)}$ 0.5

$var_{pl(1-3)}$ 0.1

$var_{pl(1)}$ 0.9
$var_{pl(2)}$ 0.7
$var_{pl(3)}$ 0.5

3.
variations over
varblend

Fixed variables:
$var_{amp(1-3)}$ 1.0
$var_{pl(1-3)}$ 0.5

Open variables:

var_{blend} 0.0

var_{blend} 0.5

var_{blend} 1.0

2.0m

ERF(Daylight)
37.3 [-]

Multi dimensional plot of
(x, Energy), (y, Cost),
(z, ERF(Daylight)) and
(u, Thermal load)
smaller numbers are better

Thermal load

2.54 [h]

344.36 [h]

sub-optimal

optimum = (0, 0, 0, 0)

GlzType2

GlzType3

Energy
180.4 [kWh/m2 year]

GlzType1

GlzType4

GlzType6

GlzType5

GlzType7

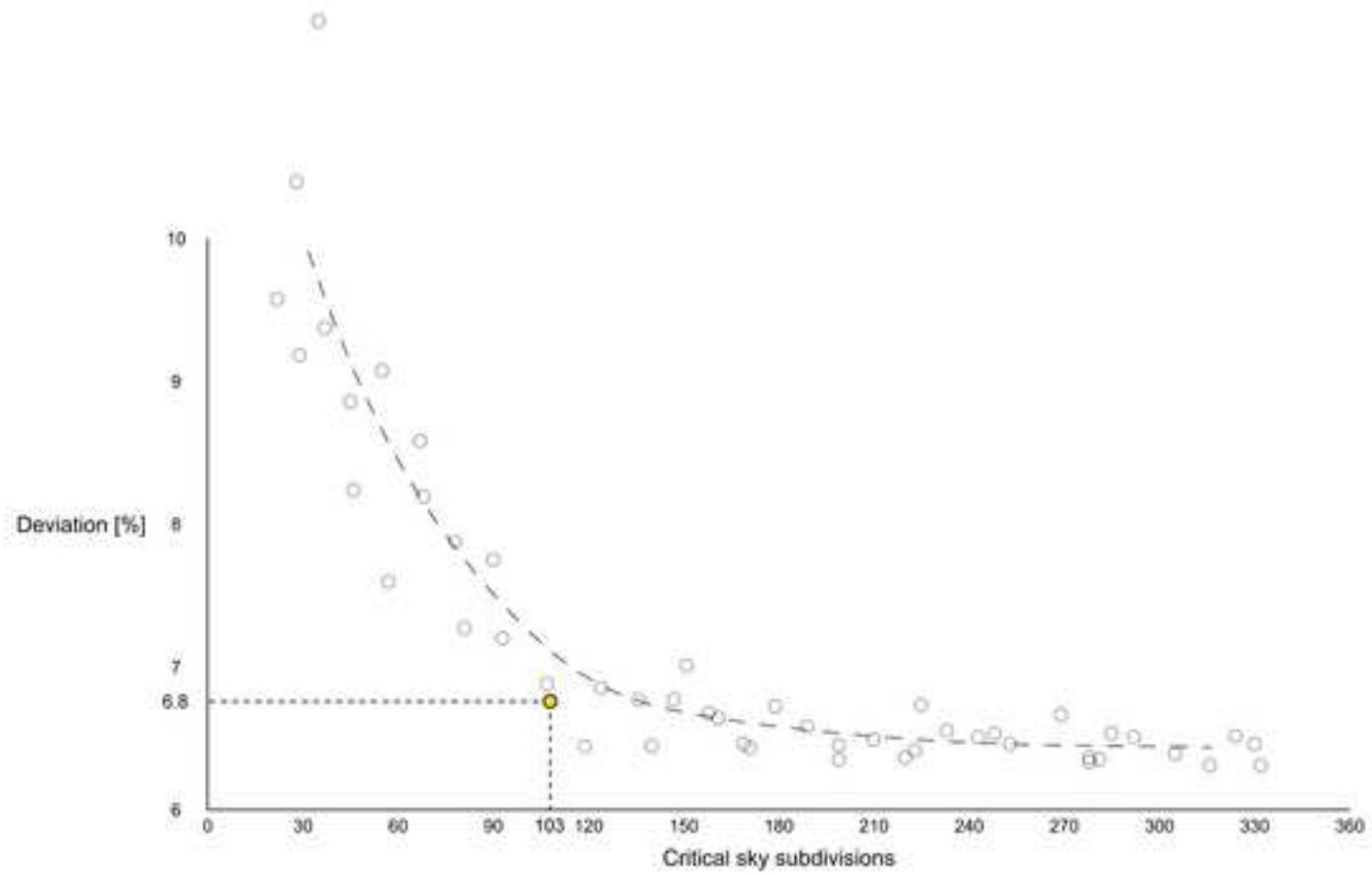Cost
134.5 [-]

Figure 7
[Click here to download high resolution image]

Solution Number	1	2	3	4	5	6	7
Energy [kWh/m² 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

Page 22 of 25