



Hybridization of an evolutionary algorithm and simulations of co-evolutionary design processes for early-stage building spatial design optimization

Sjonne Boonstra ^{a,*}, Koen van der Blom ^b, Hèrm Hofmeyer ^a, Michael T.M. Emmerich ^b

^a Eindhoven University of Technology, The Netherlands

^b Leiden Institute of Advanced Computer Science, Leiden University, The Netherlands



ARTICLE INFO

Keywords:

Optimization
Building spatial design
Evolutionary algorithm
Co-evolution
Hybridization
Multi-disciplinary
Structural design
Building physics

ABSTRACT

Three methods for early-stage building spatial design optimization are presented, demonstrated, and compared for their qualities and limitations. The first, an evolutionary algorithm, can find well-distributed approximations of the Pareto front, but it uses many design evaluations and it can only explore a limited part of the entire design search space (i.e. the collection of all possible design solutions). The second, simulations of co-evolutionary design processes, can find improved design solutions relatively fast within an unrestricted design search space, however, they typically only find discretely distributed Pareto front approximations. For the third method, hybridization is proposed to combine the first two methods into two new hybrid methods, such that their advantages are combined and their disadvantages are diminished. The methods have been applied in an initial case study, which shows that hybridization can improve search efficiency and speed, and it can search larger design search spaces.

1. Introduction

The built environment is responsible for a large part of global energy use and resource consumption, estimations of its contribution range between 40% to 60% [2,32]. For that reason, optimization in the built environment has extensively been researched and developed, but tools to optimize a design in the early stages of the building design process are still not widespread and are rarely used. Modern optimization techniques can effectively explore a design search space—i.e. the collection of all possible solutions—by strategically choosing a subset of the design search space. However, computation time increases significantly with the size of a design search space, because the amount of strategic evaluations must be sufficiently large in order to be confident about the quality of the found solutions. This makes it challenging for these modern techniques to consider large design problems all at once. In practice, building engineers approach the challenges of building design by using their knowledge, experience, and creativity, all of which are concepts that are difficult or even impossible to transfer and implement in automated optimization algorithms. This paper presents three methods that can explore the early-stage design search space of a

building spatial design: (I) an existing optimization method using a state-of-the-art evolutionary algorithm [11,15], (II) a simulation of co-evolutionary design processes, which uses design rules inspired by knowledge of- and experience with the problem formulation based on the work presented in [46,71], and (III) a hybridization of methods I and II is proposed to investigate if their advantages can be combined and their disadvantages can be diminished.

This paper is structured as follows. In Section 2, an overview of the related work is set out, and the motivation for the presented work is given. Following that, in Section 3, a toolbox that has been developed for research on early-stage building spatial design optimization is introduced, and subsequently methods I-III are presented. Thereafter, a case study to evaluate and compare methods I-III is introduced in Section 4. The results of the case study are then presented in Section 5. Accordingly, in Section 6, a discussion is given, in which the work is reviewed and critical remarks are given. Finally, the conclusion and the outlook for future work are presented in Section 7.

* Corresponding author.

E-mail addresses: s.boonstra@tue.nl (S. Boonstra), k.van.der.blom@liacs.leidenuniv.nl (K. van der Blom), h.hofmeyer@tue.nl (H. Hofmeyer), m.t.m.emmerich@liacs.leidenuniv.nl (M.T.M. Emmerich).

2. Related work and motivation

The work presented in this paper is a continuation of the work in Boonstra et al. [18]. The continued work includes among others: (a) improved simulations of co-evolutionary design processes, (b) two different hybridization schemes, and (c) a case study on the two hybridization schemes.

The remainder of this section elaborates on related research in optimization and multi-disciplinary building design (optimization), after which it is concluded with a motivation for the presented work.

2.1. Optimization

An optimization problem can be formulated with the generic mathematical expression in Eq. (1). In this formulation, the objective is to find a solution $\mathbf{x} \in \mathbf{X}$ such that it minimizes the ℓ objective functions $f_i(\mathbf{x})$. Here, a solution \mathbf{x} is a vector of v design variables: $[x_1, x_2, \dots, x_v]$, and the collection of all possible solutions \mathbf{X} is called the design search space. A solution is only considered, i.e. is feasible, when all m inequality constraints $g_j(\mathbf{x})$ and all n equality constraints $h_k(\mathbf{x})$ are satisfied.

$$\begin{aligned} \min_{\mathbf{x}} : & f_i(\mathbf{x}), \quad i = 1, 2, \dots, \ell \\ \text{subject to : } & g_j(\mathbf{x}) \leq 0, \quad j = 0, 1, \dots, m \\ & h_k(\mathbf{x}) = 0, \quad k = 0, 1, \dots, n \end{aligned} \quad (1)$$

In multi-objective optimization there seldomly exists one solution that is optimal for all objectives. The optimality of solutions is therefore assessed in terms of non-dominance. A solution \mathbf{x} is dominated by solution \mathbf{x}^* if both inequalities in Eq. (2) are satisfied. A solution is non-dominated if it is not dominated by any other solution, i.e. none of the objectives of a solution can be improved by another solution in the set without the degeneration of one or more other objectives. The set of all non-dominated solutions is called the Pareto front, and if only a subset $S \subset \mathbf{X}$ is evaluated, then the set of non-dominated solutions in S is called the Pareto front approximation (PFA). For a more comprehensive introduction on multi-objective optimization and an outline of recent developments, the reader is referred to [36].

$$\begin{aligned} \forall i : f_i(\mathbf{x}^*) & \leq f_i(\mathbf{x}) \\ \exists i : f_i(\mathbf{x}^*) & < f_i(\mathbf{x}) \end{aligned} \quad (2)$$

An optimization problem may be approached by evaluating solutions from the design search space at random. However, depending on the size and feasibility of that space, the chance that well-performing feasible solutions are selected is small. As a consequence, many evaluations are required in order for the Pareto front approximation to converge. Applying search rules on solutions—so-called heuristics—can reduce the number of necessary evaluations, because they modify a solution directed at improving that solution. Yet, heuristics interactively apply small local improvements, from which often a local optimum but not necessarily a global optimum is obtained. Many modern heuristic algorithms are instantiations of meta-heuristic search methods [43], which define a generalized structure for heuristic search. These techniques often employ randomness to introduce the required variation to escape local optima, and as such continue to search for the global optimum. Well-known examples are: particle swarm optimization (PSO) [30], where solutions are steered around the design search space by using information from both the current locally and globally best-known solutions; Or, evolutionary algorithms (EAs) [4], in which random variations are applied iteratively to a selection of solutions, the so-called parent population. The resulting solutions—the so-called offspring—are then evaluated using the objective functions and constraints and subsequently compete with the solutions of the parent population to be part of a new parent population, which replaces the old. When selecting solutions for a new population in multi-objective optimization, not only the quality of a solution but also the diversity that it adds to the population is considered.

Solutions may need to be selected from the Pareto front approximation during and after multi-objective optimization. A human expert can indicate their preferred solution, but automated selection mechanisms also exist. For example, to select the parent solutions from a population in evolutionary algorithms, or, to select the final solution that will be utilized by the user of an optimizer. Selecting one solution can be achieved by selecting the best solution in one objective, or by selecting a knee-point solution, i.e. a solution that—in objective space—lies closest to an ideal point, whether or not normalized [33]. Selecting multiple solutions at once may for instance be achieved by hypervolume-based subset selection [23,54].

To reduce computational cost and improve quality, the design search space of an optimization problem is often (implicitly) restricted in size and complexity. Such restrictions—called a superstructure [77]—prevent design variables from being added or removed from the optimization problem. Examples of works in which a superstructure is introduced to an optimization problem are found across different research fields, e.g. for flow configuration in chemical reactors [48]; for structural topologies [7]; for the dimensioning of a catamaran structure [69]; and for several different case studies [5,13]. Many state-of-the-art search methods require the problem to be superstructured, e.g. the usefulness of a gradient can be questioned when the number of variables would differ between solutions. Even though the introduction of a superstructure is in many cases useful, it may exclude global optima from the (limited) design search space, as obviously the location of these global optima within the entire design search space is not known a priori. Therefore, superstructure-free methods [77] are researched as well, e.g. for chemical process networks [35]; for finding boolean functions [29]; for finding electronic circuits [52]; for heat exchangers [28]; and for structural topologies [49,50].

2.2. Multi-disciplinary building design optimization

In the architecture, engineering, and construction (AEC) industry, optimization becomes increasingly important in reducing both the environmental and the financial footprints of designs. Literature includes—among others—methods to optimize: building envelopes for minimal heating, cooling, and lighting costs [31]; structural grillage systems for increased stiffness and less material use [16]; heating, ventilation, and air conditioning (HVAC) systems for minimal costs [66]; structural systems, while allowing user interaction during the design search space exploration [61]; and form-finding of reticulated shell structures [82]. Apart from investigating different objectives, the correct application and verification of optimization methods is also studied, for example by Hamdy et al. [45]. Furthermore, new representations for building design are researched too, e.g. equation-based models that make it possible to apply gradient-based and analytical solution search [80].

From literature it is observed that building design optimization is primarily researched per discipline and focused on sub-parts of the building design. However, due to complex trade-offs between disciplines the performance of each discipline is often compromised in order to achieve better performance in the other disciplines. Therefore, multi-disciplinary optimization has been researched, e.g. considering thermal load, usable area and cost [40]. An overview of available tools for multi-disciplinary building optimization is given by Díaz et al. [27].

In the early stages of the design process, the impact of design decisions on the performance is high and decreases rapidly as the design process progresses [78]. However, design support tools and optimization methods are predominantly available for later stages of a design process [56]. For these reasons there is a growing demand for research aimed at the support of early-stage building design (optimization) [26,62,63,65,76].

One of the reasons for research to focus on late-stage design may be that methods are often developed to be compatible with existing computer-assisted design (CAD) software, which mainly supports design processes in more advanced design stages. For example: Geyer [41]

implemented an optimization method in a Building Information Modelling (BIM) environment. Asl et al. [3] and Welle et al. [79] each present a building thermal optimization in a BIM-based design search space; Caldas [24] presents several case studies on the application of a CAD environment integrated with an evolutionary algorithm. Nonetheless, in literature it is recognized that there is a current demand for optimization in the AEC industry [64], and methods for existing CAD software are thus relevant. Therefore, Díaz et al. [27] give an overview of the challenges that need to be overcome before a practical application of optimization is possible in the AEC industry. Also Boonstra et al. [17] have looked at gaps to the practical use of optimization techniques for conceptual building design in a BIM environment. One of these challenges is user interaction, and interesting directions to overcome this challenge have been published by: Mora et al. [60], who present a framework that integrates early-stage design processes with choices regarding considerations and expectations of the final design; Steiner et al. [73], who developed a tool that interactively generates a structural system during an architectural design process; Geyer and Schlueter [42] and Schlueter and Geyer [68], who take into account user interaction with an optimization method integrated in a BIM environment; Basbagill et al. [6] and Clevenger and Haymaker [25], who each give feedback to users on the impact of the changes they made to a design, allowing them to make more informed design decisions; and Hopfe and Hensen [47], who determine the uncertainty of the effect on the performance from modifying design variables, supporting users in choosing design variables for optimization.

Software environments for early-stage design support are less prevalent but do exist, for example SEED [38]. Optimization methods for the SEED environment have been presented by Liggett [55] and Fenves et al. [37]. Despite the availability of SEED to the AEC industry, an application in practice has not been found.

Another reason for research to focus on late-stage designs may be related to the size and complexity of the design search space in the early stages of a building design process. For state-of-the-art optimization methods it is still challenging to search the entire design search space of an early-stage building design. The complexity of the design search space of early-stage building design should also be considered before a superstructure is defined, e.g. when the existence of a design variable depends on the value of another design variable. Examples of superstructures for early-stage design processes are: an application of the SEED environment to optimize building layout problems [37,55]; a genetic string from which a conceptual building spatial design is generated [72]; a unified matrix method for building spatial design [70]; or the layout of a single storey residential building in a grid [81]. These methods are developed for early-stage design and implicitly define a superstructure, for which it is not clear which designs are and which designs are not possible. A superstructure for conceptual building spatial design that has explicitly been defined by a so-called supercube is presented in [13]. Although it is not clear for all designs if they are (not) possible, there do exist obvious cases as well.

As the definition of superstructures is less straight forward for early-stage design problems, superstructure free methods are more commonly found for early-stage design, e.g. the generation of building spatial designs through shape grammars [67,74]; a framework integrating structural design and building spatial design [59]; or, simulations of co-evolutionary [58] design [19,46,71].

2.3. Motivation

Evolutionary Algorithms (EAs) can converge to a well-distributed Pareto front approximation, which can be used to gain qualitative insights in the trade-off between objectives and to study the characteristics of optimal solutions. However, EAs require a large amount of design evaluations, especially when the design search space is large, which is the case for early-stage building spatial design. Even though computational power is increasing and an EA may only need to be run once for a

static design problem, the size and complexity of an early-stage design search space is too large and complex for modern hardware, while there is a current demand for optimal design. Additionally, for each design project in practice the criteria and boundary conditions are different and they change during the design process, and as such for each project the design problem is unique and dynamic, which requires optimization to be performed multiple times during a design project. Building engineers can tackle many design problems relatively fast based on their knowledge and experience without considering many designs. Simulations of Co-evolutionary Design Processes (SCDPs)—which use design rules inspired by knowledge of- and experience with the problem formulation—have already shown that qualitatively good solutions can be obtained relatively fast [46]. However, SCDPs typically yield a discretely distributed PFA, and consequently no confidence in the quality of the found solutions can be given, i.e. there is a chance that a better design can be found in the proximity of the found solutions. Therefore, besides EA and SCDP, a hybridization of EA and SCDP is proposed to investigate if their advantages can be combined and their disadvantages can be diminished. For instance, through hybridization the speed of SCDPs may be combined with the quality offered by the PFAs found by EAs. It should be noted that this paper is focused on increasing the explorability of optimization methods in the early stages of building spatial design by means of hybridizing a state-of-the-art EA with SCDPs. The presented case study is therefore a simplification of design practice, and the inclusion of more disciplines, design variables, objectives and user interaction are left outside of the scope of this paper.

3. Methods

3.1. Building spatial design optimization toolbox

The work presented in this paper is part of a large research project on early-stage building spatial design optimization. A toolbox has been developed to support the research within this framework [20], and is also available as an open-source software repository [22]. For brevity and to avoid reiterations from previous works this subsection gives only a short introduction to the used problem representations, objective evaluations, and the constraints, accompanied with references to more detailed explanations.

3.1.1. Building spatial design representations

The design problem, i.e. building spatial design, is defined here as the determination of the dimensions and arrangement of spaces. In this work, a solution can only be composed of cuboid spaces arranged in an orthogonal grid, and as a result, spaces with curved or skewed boundaries are not possible. As such, various aspects of optimization like mutation/modification and constraints can be simplified, because special cases introduced by e.g. curved surfaces are avoided. Two representations for building spatial design have been developed: (a) the “supercube” representation, in which a three-dimensional orthogonal grid describes cells, and each of the cells can be activated for a space. Using the supercube, a space is represented by a bit mask describing cell activity, and a building spatial design is represented by the dimensions of the grid together with the bit masks of all spaces, see Fig. 1a; And (b) the “movable-sizable” representation, in which a space is defined by a location vector and a dimension vector and a building spatial design by a collection of spaces, see Fig. 1b. These representations have each been developed aimed at an application for specific optimization techniques. Evolutionary algorithms benefit from the supercube representation, because it is a superstructure, and constraints can be expressed via mathematical expressions. However, when engineers develop design rules to simulate co-evolutionary design processes, it is advantageous that they can visualize the effects of these rules. The movable-sizable representation expresses spatial information in a manner that is intuitive to engineers. A two-way conversion between the two representations has been implemented in the toolbox, a design can as such be

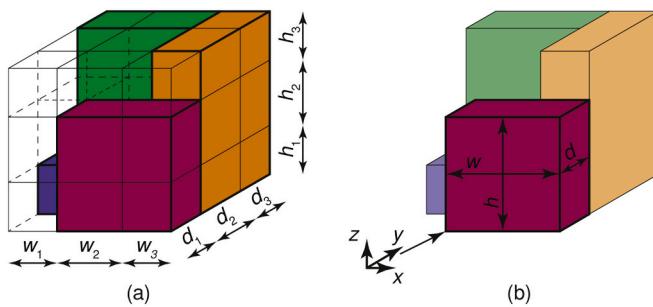


Fig. 1. Representations for building spatial designs: (a) the “supercube” representation; (b) the “movable-sizeable” representation.

expressed in any desired representation regardless of which representation was used to define it initially. For a detailed explanation of the building spatial design representations and the conversion between them the reader is referred to [20].

3.1.2. Evaluation of disciplines and objectives

Inherent to optimization is the evaluation of the objectives. A building spatial design can only be used to evaluate objectives related to the design itself, e.g. the floor area, volume, or external surface area. Objectives related to other disciplines like financial cost, environmental cost, thermal loss, or material usage cannot be extracted from a building spatial design alone. Therefore, in the toolbox, so-called design grammars have been developed that generate discipline-specific models automatically by using design rules that operate on (a part of) a building spatial design. The generated models can then be used to assess objectives for specific disciplines. Two design grammars have been developed: the structural design grammar which automatically generates a structural Finite Element Method (FEM) model; and the building physics design grammar, which automatically generates a thermal resistor-capacitor network (RC-network). The design grammars can automatically generate a structural and building physics design, which is also valuable for other aspects of building design processes [21]. The structural FEM-model can be used to compute strain energy, stresses, and displacements. The thermal RC-network model can be used to compute the heating and cooling energy that is required to keep a building within a comfortable temperature range. For more information regarding the implementations of the design grammars, the structural FEM model, and the thermal RC-network model the reader is referred to [20].

3.1.3. Constraints

To focus on feasible and functional building spatial designs, constraints can be introduced to a representation. As such, the number of spaces and the total floor area in the building can be constrained to a constant value. These constraints resemble requirements that may typically be given in a design brief to ensure the functionality of the design. Besides that: Spaces are not allowed to overlap, which is physically not feasible; Spaces should be cuboid, otherwise they are not compatible with the building spatial design representations; and the dimensions of spaces are constrained to an upper and lower bound to ensure they are practical. Moreover, as a practical approach to ensure buildings are connected to the ground, and to avoid floating spaces, all spaces must be connected to at least one other space if not on the ground, and no overhangs are allowed. Detailed information on the formulation and the implementation of constraints on the supercube representation is given in van der Blom et al. [13]. For designs in the movable-sizeable representation such checks are not needed (although possible) because constraint violating designs are avoided in the simulations of co-evolutionary design principles.

3.2. Evolutionary algorithm

Several state-of-the-art evolutionary algorithms [9] have been considered for the early-stage design of building spatial designs, which is a highly discrete Mixed-Integer Non-Linear Programming (MINLP) design problem. The NSGA-II and SMS-EMOA algorithms were evaluated in van der Blom et al. [12]. A tailored version of the SMS-EMOA algorithm, which only generates designs that comply to the constraints, was developed, configured, and assessed in van der Blom et al. [11]. And finally, a gradient ascent search method was assessed as an independent method and as a hybrid with SMS-EMOA in van der Blom et al. [15]. This paper focuses on the hybridization of the EA with simulations of co-evolutionary design processes, the reader interested in the efficiency of the EA is therefore referred to the works mentioned in this paragraph.

Based on the previously mentioned research, developments, and comparisons [11,15] the tailored SMS-EMOA algorithm has been selected to be employed for this work because it was shown to perform the best among the considered algorithms. Note that the method of choice is here selected based on median attainment curves [39], which resemble the likeliness that high-quality designs can be found by a method, and thus not necessarily the method that (by chance) found the best solution. The SMS-EMOA algorithm is not discussed in further detail here, and for detailed information the reader is referred to [34]. The SMS-EMOA algorithm has been tailored to the supercube representation, which means its initialization and mutation operators have been developed such that they only generate solutions that comply to the constraints. For more detailed information on these tailored operators the reader is referred to [11].

3.3. Simulations of co-evolutionary design processes

Simulations of co-evolutionary design processes (SCDP) are inspired by the work presented in Maher and Tang [58], in which a model for design processes is developed that takes into account co-evolutionary design principles. Co-evolution addresses the inter-dependencies that exist between the problem and a design search space. For example, in the process of designing a structural design for a building, the structural design can be optimized, but as a result it may be impossible for the building spatial design to be realized using the optimized structural design. To simulate the co-evolutionary design processes involved in building spatial and structural design an SCDP-method has been developed in Hofmeyer and Davila Delgado [46], which was shown to be effective for optimization purposes. At the top of Fig. 2 a schematic illustration of the SCDP method is given. It is started with a building spatial design (A) for which then a discipline model (e.g. a structural FEM model) is created (B). The discipline model is then optimized (C), e.g. using structural topology optimization [8] or by removing low-stressed structural elements. Accordingly, a new building spatial design is created using the optimized discipline model as a starting point (D), e.g. no spaces are placed where structural elements are less useful. Finally, because the initial building spatial design was created with certain design requirements (e.g. a number of spaces or volume), the new building spatial design is modified such that these design requirements are met. At the bottom of Fig. 2 an example of two consecutive SCDP loops has been illustrated for spatial-structural design of a building.

3.3.1. Simulation approaches

Here, two approaches to SCDP are introduced, which are based on the research presented in Hofmeyer and Davila Delgado [46]; Boonstra et al. [19]; Snel [71]. The first approach, here termed “SCDP with performance clusters”, uses a clustering algorithm to cluster spaces based on their performance, which is based on the work presented by Hofmeyer and Davila Delgado [46]. The second method, here termed “SCDP with boundary spaces”, groups the spaces that are located at the boundary of

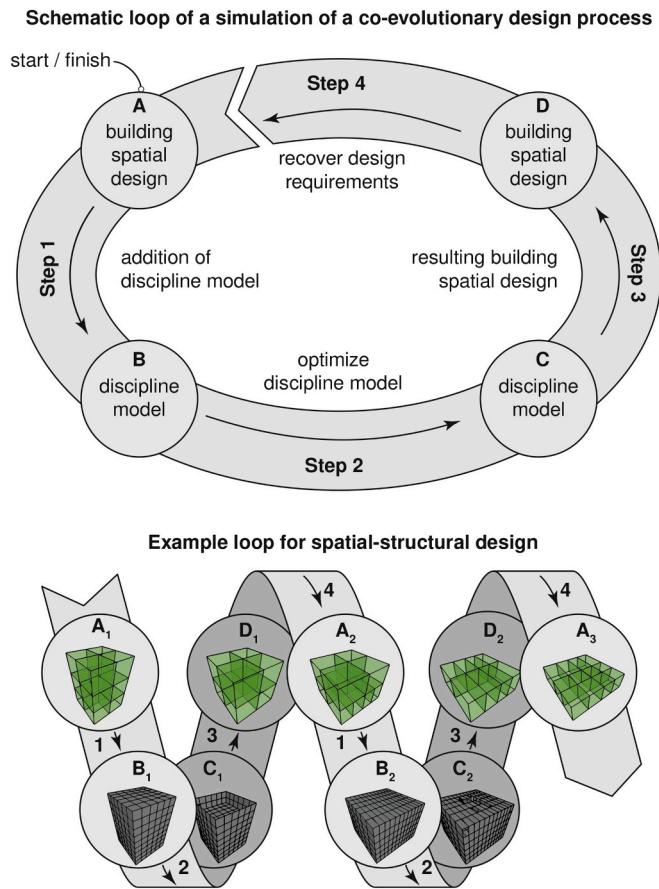


Fig. 2. Schematic loop and example of a simulation of a co-evolutionary design process.

the building spatial design for each orthogonal direction, which is based on the work presented by Snel [71]. Both approaches are introduced here, because each was found to work better for particular initial designs and objectives. For each of the two simulation approaches, a number of clusters/groups is selected based on (poor) performance, and all spaces within the selection are then removed. Accordingly, the total floor area and the number of spaces of the original building spatial design are recovered by scaling the dimensions and by splitting the remaining spaces of the newly created building spatial design. Note that this way, the SCDP method modifies a building spatial design such that the constraints (see also Section 3.1.3) remain satisfied. The SCDP method as explained in Fig. 2, is detailed here in the four steps below. Note that the third step is specified twice: once for “SCDP with performance clusters” (step 3a), and once for “SCDP with boundary spaces” (step 3b). Also note that the code used for both SCDP approaches has been made available in the open-source software repository of the toolbox Boonstra and Hofmeyer [22].

Step 1. Discipline-specific designs are generated for a given building spatial design. To that end, the design grammars are employed to generate a structural FEM model and a thermal RC-network model for the building spatial design, see also Appendix A.

Step 2. The FEM and thermal RC-network models analyzed in step 1 yielded the objective values for the whole building spatial design. For each space s in the building spatial design and for each objective t a performance value $f_{t,s}$ is computed. For the heating and cooling objective ($t = BP$) this performance value is the cumulative of the heating and cooling energy that has been simulated by the RC-network model for that space divided by the space's floor area. For the strain energy objective ($t = SD$), it is computed as the sum of strain energy over all elements that are coincident with that space (including its surfaces)

divided by the space's floor area. Once $f_{t,s}$ has been obtained for each space and each objective, it is normalized following Eq. (3). Values for $a_{t,s}$ and $b_{t,s}$ for structural performance ($t = SD$) are: $a_{SD,s} = \max_s(f_{SD,s})$ and $b_{SD,s} = \min_s(f_{SD,s})$; whereas for thermal performance ($t = BP$) they are: $a_{BP,s} = \min_s(f_{BP,s})$ and $b_{BP,s} = \max_s(f_{BP,s})$. The function for v in Eq. (3) is given in Eq. (4), which switches to a linear scaling for thermal performance ($t = BP$) and to a log-linear scaling for structural performance ($t = SD$). In Eq. (4), c is a constant that increases the resolution of values around zero, although in this work no negative or values close to zero were normalized a value of $c = 150$ is used. Here, the definition of a poor performing space is that space for which its normalized performance lies closest (in Euclidean space) to the dystopian point ((1, 1) in case of two normalized objectives), and similarly a well-performing space relates to the utopian point ((0, 0) in case of two normalized objectives). Note that in this way, a space with a high heating and/or a high cooling demand is labeled as poor performing with regards to thermal design. Whereas a space with a low amount of strain energy is labeled as poor performing with regards to structural design, which may be perceived as counter-intuitive because the objective is to minimize strain energy. However, this notion has been observed to work well in Snel [71] and it can be supported from the point of view of proportional topology optimization [10], where a structural topology is optimized by explicitly adding material at places where it is needed most (i.e. locations that deteriorate the objective) and removing material where it is not needed (i.e. locations that do not contribute to the objective). From that perspective, if a space associated with low strain energy is removed, this can be interpreted such that the structural material that realizes the space is not in the optimal location with respect to minimizing the structural objective.

$$\hat{f}_{t,s} = \frac{v_t(b_{t,s}) - v_t(a_{t,s})}{v_t(b_{t,s}) - v_t(f_{t,s})} \quad (3)$$

$$v_t(u) = \begin{cases} u & t = BP \\ \text{sgn}(u) \cdot \log(1 + u \cdot 10^c) & t = SD \end{cases} \quad (4)$$

Step 3a. (SCDP with performance clusters). Spaces are clustered by their normalized performance, and accordingly the spaces in one or more clusters are removed from the building spatial design. Here clustering is performed using the k-means algorithm [57], which groups the spaces into k clusters. To reduce the sensitivity to stochastic initialization, the algorithm is run l times per cluster size k , after which the best clustering is chosen based on the lowest sum of cluster variances, where a cluster's variance is the averaged sum of squared distances between each data point and the mean of the cluster. Moreover, because a priori (and without supervision) it is not known which cluster size is suitable for the problem, a range of cluster sizes $[k_{\min}, k_{\max}]$ is defined. To select a suitable cluster size k the method presented by Krzanowski and Lai [53] is used. When a suitable clustering of the spaces has been computed, the spaces in the cluster with a mean that lies closest to the dystopian point (i.e. (1, 1)) are removed. It is then checked if at least 15% of the spaces has been removed from the building spatial design. If this is not the case, all spaces in the next cluster for which the mean lies closest to the dystopian point are removed. This is repeated until at least 15% of the spaces has been removed and this ensures that the design is significantly modified.

Step 3b. (SCDP with boundary spaces). A group of spaces that is located at the boundary of a building spatial design is removed from the building spatial design. For this, six selections S_p (where $p \in 1, 2, \dots, 6$) of spaces that are located at the boundary of a building spatial design are made, one for each orthogonal direction $\mathbf{n}_p \in \{+\hat{\mathbf{i}}, -\hat{\mathbf{i}}, +\hat{\mathbf{j}}, -\hat{\mathbf{j}}, +\hat{\mathbf{k}}, -\hat{\mathbf{k}}\}$, where $\hat{\mathbf{i}}$, $\hat{\mathbf{j}}$, and $\hat{\mathbf{k}}$ are the unit vectors in x -, y -, and z -direction respectively. For each direction a selection is made as follows, see also Fig. 3. First the extreme coordinate $c_{p,\text{extr}}$ in the corresponding direction is searched. Second, among the spaces that contain $c_{p,\text{extr}}$, the maximum dimension $d_{p,\text{max}}$ is searched. Third, a so-called selection plane is defined by a normal, i.e. the direction \mathbf{n}_p , and the point P_0,p , which is

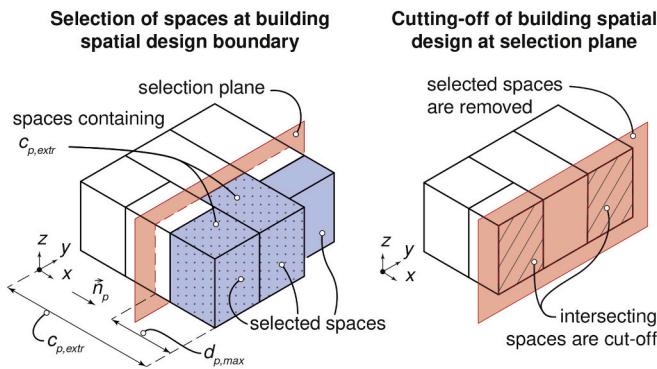


Fig. 3. Selection of spaces into façade groups and cut-off of such a group.

defined by the position vector $\mathbf{p}_{0,p} = [0 \ 0 \ 0]^T + c_{p,extr} \cdot |\mathbf{n}_p| - d_{p,max} \cdot \mathbf{n}_i$. Finally, the selection of spaces in direction \mathbf{n}_i includes all spaces of which each point P_q (defined by position vector \mathbf{p}_q) inside or on the space's boundary satisfies the following condition: $\mathbf{n}_p \cdot (\mathbf{p}_q - \mathbf{p}_{0,p}) \geq 0$; in other words, each space that is completely on the side of the selection plane (the side in the direction of its normal). A selection S_p is disregarded if it includes all the spaces within a building spatial design. Then, for those selections that remain, an average performance $\mu_{f_{i,s}}$ is calculated by averaging the performance of each space in the selection over the number of spaces in the selection, see Eq. (5), where $n_{S,p}$ is the number of spaces in selection S_p . Accordingly, the selection plane that was used to make the selection with the average performance that lies closest to the dystopian point (i.e. (1, 1)) is used to cut-off that part of the building. This is achieved by first removing those spaces from the building spatial design that are included in the corresponding selection. Following that, the spaces that intersect the selection plane have their dimensions and coordinates (if affected) reduced such that the space is cut-off at the selection plane, see Fig. 3. It is then checked if any of the cut spaces violate the lower bounds of a constraint on the dimensions of a space. If this is the case, the building spatial design is extruded at the cut-off plane in the direction of its normal until it satisfies all constraints.

$$\mu_{f_{i,s}} = \frac{\sum_{s \in S_p} f_{i,s}}{n_{S,p}} \quad (5)$$

Step 4. The floor area and the number of spaces in the modified building spatial design are restored to their initial values. First, the floor area is scaled by multiplying the x - and y -coordinates of each space by a factor of $\sqrt{A_0/A_{mod}}$, where A_0 is the initial floor area and A_{mod} is the floor area of the modified building spatial design. After scaling, the coordinates of spaces are rounded to the nearest whole millimeter to prevent small overlaps and gaps between spaces due to numerical errors. Second, until the initial number of spaces has been obtained, a space is selected from the building spatial design to be split into two new spaces. A space is selected for splitting if its largest dimension is also the largest among all space dimensions in the building spatial design. However, to prevent a constraint violation, splitting is not performed if the selected dimension is less than twice the lower bound of the constraint on that dimension. In such cases the next space that matches the criteria is selected for splitting. If a space can be split, it will be split across its center with a cutting plane perpendicular to the direction of the dimension through which the space was selected. Finally, each coordinate value in the building spatial design is rounded to the nearest multiple of 100 mm, this prevents disproportional geometric ratios in the models generated by the design grammars, which may cause numeric errors. For instance a structural flat shell element of 1 mm wide and 3000 mm high would result in a ratio of 1:3000, which is highly unlikely to yield accurate results.

3.4. Hybridization

A comprehensive taxonomy of hybridization schemes is presented by Talbi [75]. From this taxonomy, two promising hybridization schemes are selected: the high-level relay hybrid and the high-level teamwork hybrid. Where in a high-level hybridization scheme, each used method is self-contained, and in a low-level hybridization scheme each method is integrated with the other method(s). Although the low-level scheme is interesting, the integration between the methods brings along several considerations, for instance how to deal with a changing design search space while the EA is running. Therefore, because this work entails an initial study, only the high-level scheme will be investigated for the hybridization of the EA and SCDP. A relay hybrid employs each method in sequence, whereas a teamwork hybrid employs each method in parallel. Because there is no prior indication that the two methods would benefit from the relay scheme or from the teamwork scheme, both hybridization schemes are used here.

3.4.1. Relay hybridization

The relay hybridization is illustrated in Fig. 4. The scheme consists of iterations in which the EA and the SCDP methods are run consecutively. Apart from the settings required for each individual method, the relay scheme requires the definition of an initial supercube, a budget for the total number of design evaluations n_{tot} , a budget for the number of exploratory evaluations n_{expl} , an evaluation budget for each exploratory EA n_{EA} , and a number of simulation loops for each SCDP run n_{SCDP} . An iteration starts with ten runs of the EA, after which three designs are selected from the resulting overall Pareto front approximation: the design with the best structural performance (SD); the kneepoint design (KP); and the design with the best thermal performance (BP). These are selected by first normalizing the PFA following Eqs. (3) and (4), accordingly the selection consists of those designs that have their normalized performance closest to the following points: SD, (0, 1); KP, (0, 0); and BP, (1, 0). Thereafter, the SCDP method is applied with $n_{SCDP, set}$ different settings, and each simulation is run for a total of $n_{SCDP, loop}$

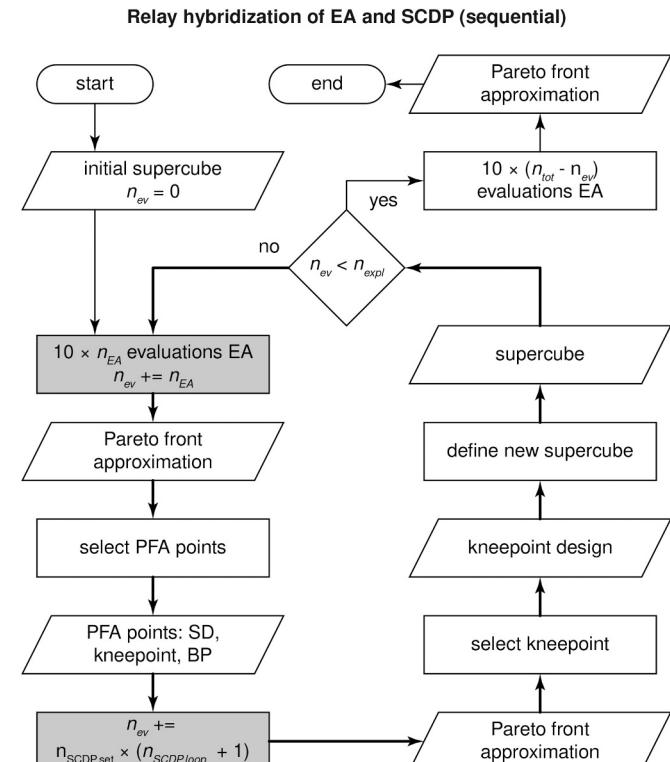


Fig. 4. Flow chart of the relay hybridization.

loops, which yields a total of $n_{SCDP, eval} = n_{SCDP, set} \cdot (n_{SCDP, loop})$ evaluations performed by SCDP per iteration. Here, an iteration starts with the EA and not with the SCDP methods, because that would require the definition of initial designs. A definition of initial designs is avoided to prevent a bias that may be introduced by the input, for instance: the initial design may be chosen such that it is already an optimum, or, the initial design may be deliberately be chosen such that it performs bad to avoid it from being an optimum. From all designs that are found by the SCDP methods, the non-dominated points are selected (i.e. the PFA), normalized, and consequently the kneepoint design is selected in the same approach as described for the PFA of the EA. The kneepoint design is then used to define a new supercube as follows. The kneepoint design is converted from the movable and sizable representation to the supercube representation (for representations see Section 3.1.1), which yields a new supercube. Then, to ensure the supercube is not too small or too large to be navigated by the EA, its size is modified using the factor η_{sc} , which is given by Eq. (6), where $n_{cell, init}$ is the number of cells in the initial supercube and $n_{cell, kp}$ is the number of cells in the supercube obtained from the kneepoint design that is obtained by the SCDP methods. The number of grids in each direction $p \in \{x, y, z\}$ (width, depth, and height) is modified following Eq. (7), where $n_{grid, upd, p}$ is the updated number of grids in direction p and $n_{grid, kp, p}$ is the number of grids of the supercube in direction p obtained from the kneepoint design. A supercube is scaled in this way to ensure that the ratio between the number of cell grids in each direction stays more or less the same. This way if a well-performing design solution is tall and narrow then it is reasoned here that an appropriate supercube for that design is also tall and narrow. Note that in this way, the amount of cells that can be defined by a supercube is limited to an upper bound, which is defined implicitly by the initial definition of a supercube. After the supercube has been updated, the iteration is finished, a new iteration is only started if the number of elapsed evaluations n_{ev} is smaller than n_{expl} . If no new iteration is started, the EA is employed for ten more runs, each with an evaluation budget of $n_{tot} - n_{ev}$.

$$\eta_{sc} = \frac{n_{cell, init}}{n_{cell, kp}} \quad (6)$$

$$n_{grid, upd, p} = \begin{cases} \eta_{sc} \cdot n_{grid, kp, p} & \text{if } \eta_{sc} \cdot n_{grid, kp, p} > 1 \\ 1 & \text{otherwise} \end{cases} \quad (7)$$

3.4.2. Teamwork hybridization

The teamwork hybridization is illustrated in Fig. 5. Many of the processes are the same as these used in the relay hybridization and are therefore not explained here again. The teamwork hybridization requires the same parameters to be defined: an initial supercube, n_{tot} , n_{EA} , $n_{SCDP, set}$, and $n_{SCDP, loop}$. Each iteration is started by simultaneously running the EA and the SCDP methods, after which the resulting Pareto front approximations are merged, i.e. the non-dominated solutions are selected from the combined solutions of both the EA and the SCDP methods. In order to avoid the definition of initial design(s) for the SCDP methods, the first iteration excludes the SCDP methods. Accordingly, the best structural design (SD), the kneepoint design (KP), and the best thermal design (BP) are selected, which serve as input for the SCDP methods in the next iteration. Then, based on the kneepoint design a new supercube is defined, which serves as input for the EA in the next iteration. The iteration is then finished, and a new iteration is only started if the number of elapsed evaluations n_{ev} is smaller than n_{expl} . If no new iteration is started, the EA is employed for ten more runs, each with an evaluation budget of $n_{tot} - n_{ev}$.

Teamwork hybridization of and SCDP (parallel)

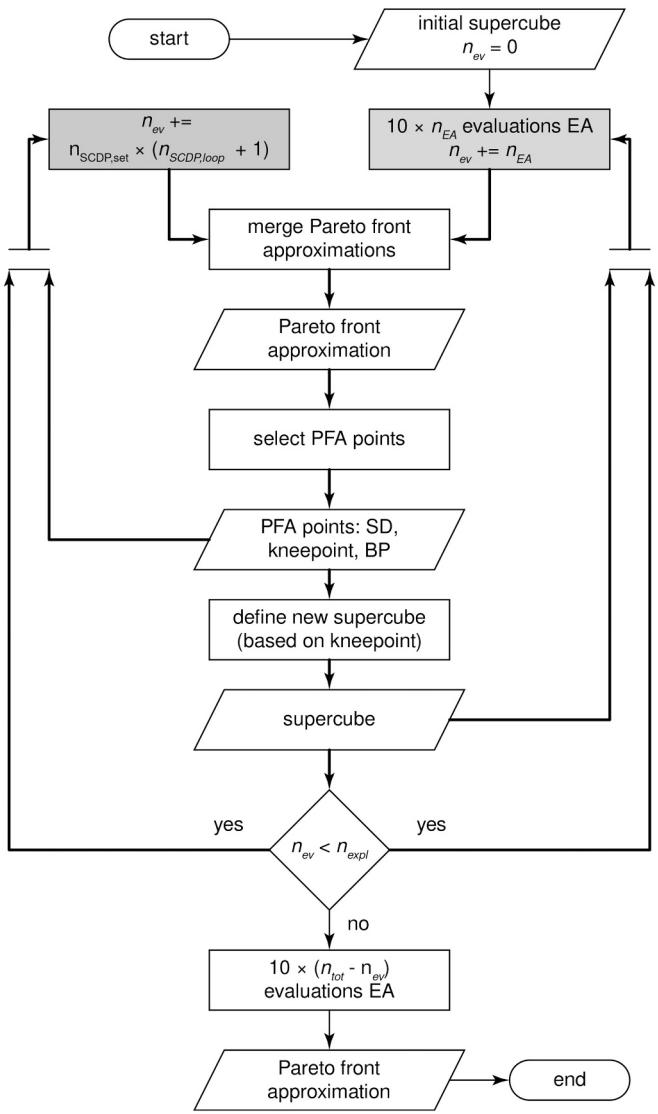


Fig. 5. Flow chart of the teamwork hybridization.

4. Case study

4.1. Design problem

4.1.1. Objectives

Two objectives are defined: a minimal total sum of strain energy [N mm] analyzed by the FEM model, and a minimal total amount of heating and cooling energy [kWh] simulated by the RC-network model. The sum of strain energy is calculated over each element and each load case in the FEM model. Minimizing strain energy relates to minimizing flexibility or maximizing stiffness, and it is a common objective for structural topology optimization. The total amount of heating and cooling energy in a building during the period that is simulated by the RC-network model is calculated as the cumulative energy spent on keeping the temperature of each space between a lower and an upper bound.

4.1.2. Constraints

The constraints regarding the number of spaces and the floor area (see also Section 3.1.3) are as follows: the total number of spaces is exactly 50, and the total amount of floor area is exactly 750 m². Moreover, the dimensions of each space are constrained: in z-direction to a

value within a range of 3 m to 20 m, and in both the x- and y-directions to values within a range of 0.5 m to 20 m.

4.2. Settings

4.2.1. Design grammars

The settings for the design grammars and the evaluation of the discipline-specific models that will be used in this work (see also Section 3.1.2) are adopted from [15], however for completeness, a description of these settings and the ensuing models is given in Appendix A. The structural design grammar generates a structural FEM model for a building spatial design by using the settings in Appendix A.1, which also describes the resulting structural FEM model. Similarly, the building physics grammar generates a thermal RC-network model by using the settings in Appendix A.2.

It should be noted that these settings generate only a single structural FEM and a single RC-network model for a building spatial design. Moreover, the models are generated based on conceptual building spatial designs, and based on assumptions regarding late-stage design decisions. Therefore, the evaluations found by these models cannot be considered quantitative, however, they can be used for the qualitative comparison of the structural and thermal behavior between building spatial designs. Such a qualitative comparison will become more reliable if they are based on the evaluation of multiple different variants generated by different design grammars, but this will come at the cost of computation time. Additionally, in previous studies [14,19,46,71] it has been observed that evaluations based on one variant do lead to improvements in the building spatial design that can also be explained from an engineering point of view. For instance, for structural design, low-rise building spatial designs without long spans were generally found to be optimal, and for thermal building physics, a square floor plan was found to be optimal (which is the case for orthogonal building spatial designs).

4.2.2. Evolutionary algorithm settings

For the case study, a supercube with a grid size of $6 \times 6 \times 6$ cells (i.e. 216 cells) for 50 spaces has been used. As such, a solution is represented by $6^3 \cdot 50 = 10800$ binary design variables (i.e. bitmasks of spaces) and $3 \cdot 6 = 18$ continuous variables (i.e. grid dimensions). The evaluation budget of the algorithm is set to 5000 evaluations, of which the first 6 solutions are the initial population. Other settings for the algorithm are adopted from [11], and are here given in Appendix B. With these settings, the algorithm is run ten times, which is to avoid a large dependency on the stochastic initialization and modification of solutions. It should be noted that, prior to the work presented in this paper, the tailored SMS-EMOA algorithm was not applied to problems exceeding a supercube containing 100 cells and 5 spaces. Moreover, this work uses for the first time a floor area constraint with the SMS-EMOA algorithm, which replaces the volume constraint that is used in previous works. This floor area constraint is implemented in a similar way as the volume constraint in [11], but with a higher accuracy. A detailed explanation of the floor area constraint is given in Appendix B. Considering that the adopted configuration for the method [11] has been configured for a smaller supercube size in combination with a volume constraint, there may exist a configuration for the method that is more suitable to the problem presented in this paper. However, this is not further investigated here.

4.2.3. Simulations of co-evolutionary design processes settings

Both SCDP with performance clusters and SCDP with boundary spaces are applied to three predefined building spatial designs, which are given in Fig. 6. Design 1 is a ten storey tower with five spaces on each floor; Design 2 is a five storey apartment building with ten spaces on each floor; and Design 3 is a one storey building with 50 spaces. These designs are selected/designed because of their different characteristics, as such the different SCDP approaches are validated for different starting

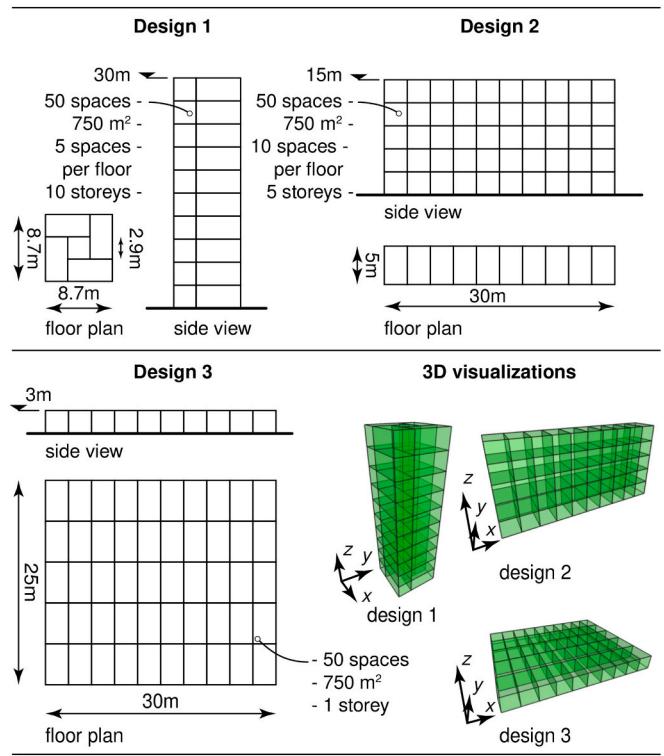


Fig. 6. Designs that serve as an initial design for SCDP.

points. For each design and each SCDP method, ten loops are simulated. Moreover, for each design and each SCDP method, three separate runs are performed: (i) evaluating only the structural (SD) objective in step 2; (ii) evaluating only the thermal (BP) objective in step 2; and (iii) evaluating both the structural (SD) and the thermal (BP) objective at the same time in step 2. With three different designs, three different evaluation methods, and two different SCDP approaches, these settings define $3 \cdot 3 \cdot 2 = 18$ simulations of a design process. Each simulation consists out of ten loops, and thus—including the evaluation of the final design—each simulation uses $(10 + 1)$ design evaluations (i.e. one FEM analysis and one RC-network simulation). Therefore, in the case study SCDP uses a total of $(10 + 1) \cdot 18 = 198$ design evaluations overall. Finally, for clustering in step 3a, the following range of possible cluster sizes is used $k_{min} = 2$ and $k_{max} = 10$, and for each cluster size k the k-means algorithm is run $l = 50$ times to reduce a possible sensitivity to the stochastic initialization of a clustering. The settings above have also been summarized in Table 1.

4.2.4. Hybridization settings

The settings of the EA and SCDP that overlap with the hybrid method are the same as specified in Section 4.2.2 and Section 4.2.3 respectively. The remaining settings of the hybrid method are the same for both schemes, and are as follows: the initial supercube is set to a supercube of

Table 1
Settings used for SCDP.

Setting	Value(s)
Initial design	{Design 1, Design 2, Design 3}
Evaluation*	{SD, BP, SD & BP}
SCDP approach	{performance clusters, boundary spaces}
Number of loops	10
Min. clusters k_{min}	2
Max. clusters k_{max}	10
K-means runs l	50

* The objective value(s) that are used to evaluate a space, see also step 2 in Section 3.3.1.

$6 \times 6 \times 6$; the total evaluation budget is set to $n_{tot} = 5000$; the exploratory evaluation budget is set to $n_{expl} = 1200$; the evaluation budget of each exploratory EA is set to $n_{EA} = 500$; the number of simulation loops for SCDP is set to $n_{SCDP, loops} = 10$; and, the number of settings for SCDP is set to $n_{SCDP, set} = 18$, which follows from the settings used for the SCDP method in Section 4.2.3. Note that n_{expl} is chosen such that at least two full iterations of both hybrid methods are completed. The settings described above have also been summarized in Table 2.

5. Results

The results found by each method for the case study (see also Section 4) are presented in this section. Moreover, this section is concluded with a comparison of the found results (Section 5.4).

5.1. Evolutionary algorithm results

Fig. 7 shows a graph with the results of all the solutions that are considered over all 10 runs of the evolutionary algorithm together. In the graph, each dot represents the performance of a solution, and the gradient of a dot corresponds to the ordinal number of the evaluation of a solution within a run, which gives insight into the birth time of solutions, i.e. the amount of evolutionary operations that have been performed on the population before a solution was found. Moreover, the blue triangles are the solutions—over all 10 runs—that are non-dominated (i.e. the Pareto front approximation) after all 5000 evaluations have been completed, whereas the red circles are the solutions—over all 10 runs—that are non-dominated after only the first 500 evaluations have been completed. Note that, here the best solutions out of all 10 runs are selected for the PFAs, and there is no confidence that a similar performance can be found again from a single run of the EA. Therefore it may be more appropriate to compute the median attainment curve [39] over all 10 runs, which does provide confidence of what a single run of the EA is likely to achieve. Nevertheless, in this paper the EA is used in a competitive setting and the best out of all 10 runs is deemed appropriate. Finally, it should be noted that, to better visualize the results, solutions with a performance outside the 95th percentile are not plotted in the graph.

On the right of Fig. 7, solutions that correspond to specific points on the PFAs have been visualized. Within each PFA, these points are: the solution with the least amount of strain energy (structural design, "SD"); a trade-off between the two objectives (knee-point "KP"); and, the solution that requires the least amount of energy to retain a comfortable temperature (building physics "BP"). Here, the knee-point has been selected as the solution for which its normalized performance is the closest to the origin (0, 0) when each objective value is normalized to a [0, 1] interval between the minimum and maximum found values (among the PFA solutions) for the corresponding objective. Additionally, it should be noted that strain energy is normalized on a logarithmic scale, whereas the heating and cooling energy is normalized on a linear

scale, following Eqs. (3) and (6) in Section 3.3. This way, the normalization will project a solution that performs well for the thermal objective but poorly for the structural objective closer to the origin, which is appropriate because the order of magnitude between the objectives differ. These visualizations are given for both the PFA after 500 evaluations (top right) and the PFA after 5000 evaluations (bottom right).

From the plot, on the left of Fig. 7, it can be observed that early found solutions (light grey dots) are located relatively close to the Pareto front approximation (blue triangles). This is underlined by also plotting the PFA (red circles) that is found if each run would be finished after only the first 10% (i.e. 500) of evaluations are completed. The chosen evaluation budget of 5000 evaluations thus appears to be sufficient to let the PFA converge.

When studying the visualized solutions on the right of Fig. 7, it can be noticed that each design is represented by activating almost all of the cells within the used supercube. As a consequence, each of the visualized designs is a six storey building spatial design and—for example—no two storey building spatial designs are found, even though such designs are possible with the used supercube and they are known to perform better than the tall building spatial designs that were found for the used structural objective. This suggests that the choice of supercube is not only relevant for the designs that are excluded by definition, but it is also relevant for the likeliness that certain designs are not found, e.g. a two storey building spatial design. This is likely to be caused by the used initialization and mutation operators of the tailored SMS-EMOA algorithm, because a design is initialized by 'filling up' a supercube with spaces that consist of an arbitrary amount of supercube cells, thereby checking if enough cells remain to realize all the required spaces of a design. If a relatively high amount of spaces should be realized with a relatively low amount of supercube cells, consequently (almost) all cells will be used to realize an initial building spatial design. Additionally, the mutation operators search for new designs by expanding and contracting spaces. However, spaces that consist of one supercube cell may not be contracted, whereas this may be necessary in order to expand a neighboring space. A set of interlocking spaces may then make it difficult to find new designs. Although these phenomena have been taken into account in the development of the tailored SMS-EMOA algorithm and it has been observed to efficiently explore a small supercube [11], it appears that the tailored operators cannot effectively explore a design search space that is defined by larger supercubes. Nevertheless, it should also be noted that without the tailored operators, especially in large supercubes, stochastic initialization and mutation operators are not likely to find feasible solutions at all, which has been the motivation to develop the tailored operators in the first place.

Another observation made from the visualizations of the Pareto front approximation at 5000 evaluations, Fig. 7, is that spaces that consist of more than one cell in z-direction (height) are generally narrow. For the thermal objective, tall spaces have a relatively large volume per floor area, which consequently leads to a relatively high heating and cooling demand per floor area. Because the total floor area of a building spatial design is constrained, a tall space contributes more to the degradation of the thermal objective compared to a space that is not tall. This is also observed in the visualized designs, where the structurally optimal (SD) design has a more evenly distributed grid compared to the narrow grids in the kneepoint design (KP) and the thermally optimal design (BP). In the case where a tall space exists in a building spatial design it is beneficial for the thermal objective that that space is narrow. Although an absence of tall spaces would improve it even more, such an absence has not been observed. This could originate from the limited exploration capability of the tailored SMS-EMOA algorithm, as has been discussed in the preceding paragraph.

5.2. Simulations of co-evolutionary design processes results

The results of the SCDP method are given in Fig. 8. On the left of the figure, each found solution is plotted in a graph per SCDP method and

Table 2
Settings used for both the relay and teamwork hybridization schemes.

Setting	Value(s)
Initial supercube	$6 \times 6 \times 6$
n_{tot}	5000
n_{expl}	1200
n_{EA}	500
$n_{SCDP, set}$	18 *
$n_{SCDP, loop}$	10

* The number of different SCDP settings follows from the number of designs (KP, SD, and BP), the number of SCDP approaches used (performance clusters and boundary spaces, and the number of evaluations approaches (SD, BP, SD & BP): $3 \cdot 2 \cdot 3 = 18$.

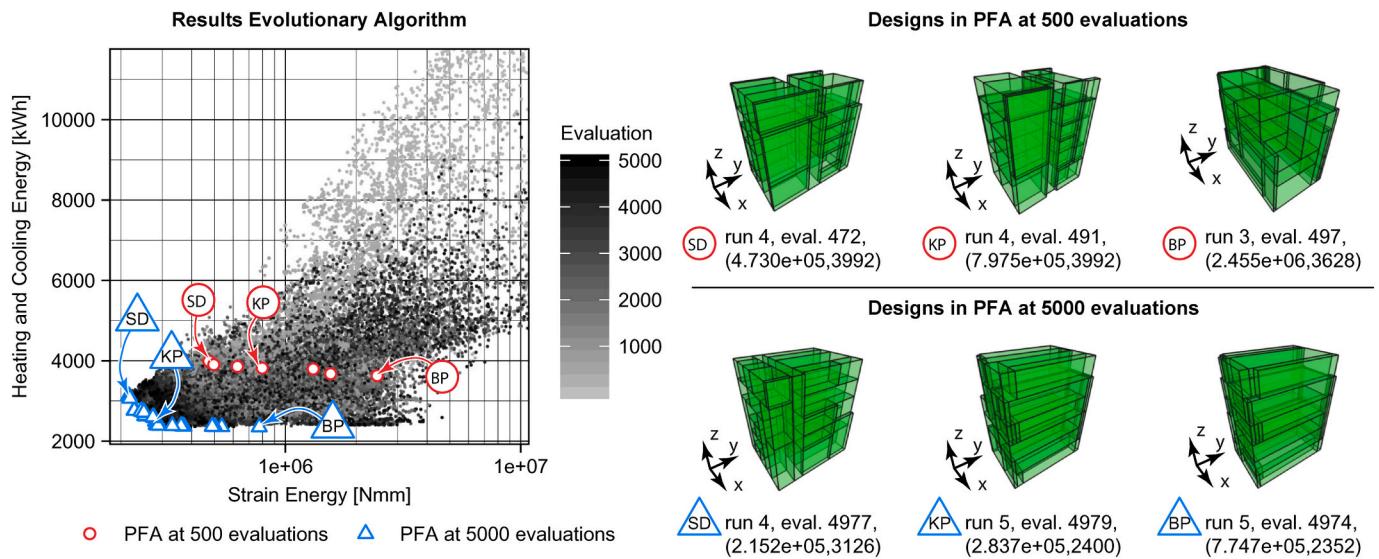


Fig. 7. Results of the evolutionary algorithm for the demonstrative example.

per design. In each graph, the performance of the initial design is plotted with a large black dot. The performance of the designs that follow from evaluating only the thermal (BP) objective is plotted with red circles, and the order in which the SCDP method found these designs is indicated with solid red lines. Similarly, the performance and the finding order of designs that follow from evaluating only the structural (SD) objective is plotted with blue squares and dashed blue lines. And, the performance and the finding order of designs that follow from evaluating both the thermal (BP) and structural (SD) objectives at the same time is plotted with black diamonds and dotted black lines.

On the right of Fig. 8, the design paths of three SCDP runs are visualized, where a design path is the collection of designs that were found by an SCDP method in the order in which they were found. For Designs 1 and 2, the design path that yielded the kneepoint has been visualized, which for both designs is the design path obtained from SCDP with boundary spaces evaluating only the thermal (BP) objective. For Design 3, no significant improvement could be made to the initial design, therefore an arbitrarily selected design path is visualized, namely that of SCDP with performance clusters evaluating both the structural (SD) and thermal (BP) objectives at the same time. Moreover, for clarity, the performance of the final design of the design path that has been visualized for each design is indicated with a yellow star in the plot that corresponds to the used SCDP method.

The graphs on the left of Fig. 8 indicate that both methods find improved solutions for both Design 1 and 2. However, for Design 3 no improved solutions are found, even though an improvement is possible since the SCDP runs on Designs 1 and 2 yield designs with better performance, all of which are two storey buildings. In order for Design 3 to be improved it should thus become a two storey building. However, this cannot be achieved when using the used SCDP method and the dimension in z-direction (height) of each space is less than 6000 mm (splitting such spaces would lead to constraint violations), hence Design 3 is a local optimum. Considering that only the shape ratio of the floor plan and the spaces can be changed using the SCDP methods, it can be concluded that Design 3 initially starts with a (locally) optimal layout. A nearly square floor plan reduces the surface area of the external area, minimizing heat losses and gains. Relatively short spans in one direction ensure minimal strain energy. These characteristics can also be observed in the visualized design paths of Designs 1 and 2. SCDP on Design 1 yields the design with the best thermal performance over all SCDP runs, which is square, however, also the shape ratios of its spaces are square. SCDP on Design 2, on the other hand, yields the design with the best structural performance over all SCDP runs because its spaces have one

relatively short span. However, the floor plan is not square, which comes at the cost of thermal performance.

Although SCDP with boundary spaces using an evaluation of only the thermal objective has yielded the best result for both Design 1 and 2 as initial design, it should be noted that the other settings for both SCDP methods have yielded designs with similar performance. Moreover, it can be observed that along the design path of any of the 18 SCDP runs, the last design is not always the best design. In fact, after reaching an improved design, the SCDP method may degrade the performance, which mostly appears to be the case when the structural (SD) objective is considered.

5.3. Hybridization results

The results of the relay hybrid are given in Fig. 9, and for the teamwork hybrid in Fig. 10. In the graphs, the performance of each EA solution is plotted with a black dot in the left graph, and the performance of each SCDP solution is plotted with a black diamond without a fill in the right graph. PFA points are plotted with the following colored shapes with a white fill: red circles, blue triangles, yellow diamonds, and purple squares. Moreover, for comparison, the PFA of the results of the EA in Section 5.1, Fig. 7 is plotted with solid grey triangles. For each SCDP plot, the design path that has lead to the kneepoint design is plotted with solid red lines. Note that for each graph the same ranges and scales are used for the axes, and that the strain energy is plotted on a logarithmic scale. Finally, the visualizations of the designs that are selected by each hybrid method are given besides the plots: in Fig. 9, in the middle the best structural (SD), kneepoint (KP), and the best thermal design (BP) from the PFA of the EA are visualized and on the right the kneepoint (KP) design from the PFA of SCDP is visualized; in Fig. 10, in the middle the best structural (SD), kneepoint (KP), and the best thermal design (BP) from the merged PFA of both the EA and SCDP are visualized.

The characteristic narrow spaces that can be observed in the designs found in the EA demonstration in Section 5.1, Fig. 7, can also be observed in the designs found by the EA runs of the hybrid methods. Even with the one storey building spatial designs in the final iteration of the teamwork hybrid, in Fig. 10, narrow spaces can be observed. This is likely caused by the empty cells in the proximity of these narrow spaces, which cause a recess in the façade. This recess increases the surface area, and as a result degrades the thermal objective. In order to minimize the effect of this recess, its depth is minimized, which is achieved by the EA by minimizing the cell grid dimensions. As a consequence, the cells

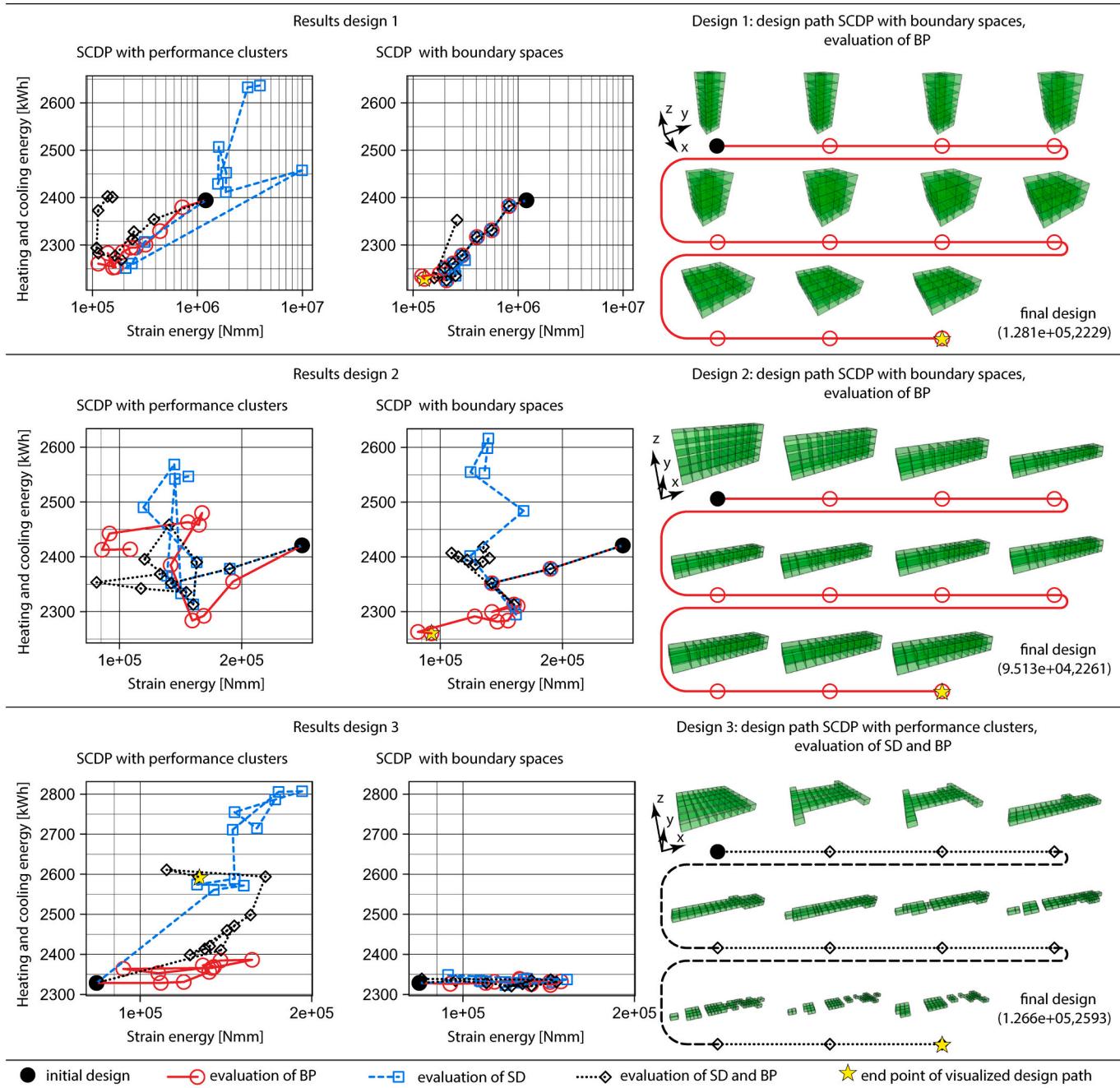


Fig. 8. Results of the SCDP runs.

that are activated within these minimized cell grids also become narrow, and as such it leads to narrow spaces. The narrow spaces in a one storey building spatial design may very well benefit the strain energy objective as well because a narrow space has a short span. Nevertheless, the EA's ability to resolve the relatively inefficient narrow spaces that cover multiple storeys could still be improved.

Furthermore, when comparing the relay hybridization with the teamwork hybridization with respect to the performance of solutions, both methods find PFAs that are close to each other. Although, small differences can be seen, the relay hybridization performs better in the structural objective, whereas the teamwork hybridization performs better in the thermal objectives. The cause of this difference can be explained from the fact that the relay hybridization finishes with a supercube of one cell in height, forcing all designs to be a one storey building spatial design, which is beneficial for the structural objective.

On the other hand, the teamwork hybridization finishes with a supercube of two cells in height, enabling two storey building spatial designs, allowing the surface area of the roof and ground floor to be halved, which reduces thermal losses and gains and it is thereby beneficial for the thermal objective. However, this difference is likely to be coincidental and not a consequence of the used hybrid method, but rather a consequence of the stochastic characteristic of the EA, as it leads to different initial designs for the SCDP methods.

Finally, also the anytime performance is compared here, which is the performance at any given moment during the runtime of an algorithm. When comparing the anytime performance, the relay hybridization is performing best, because the performances of the designs found by the relay hybrid after 5198 evaluations are similar to the performances of the designs found by the teamwork hybrid after 10,198 evaluations. This is mainly because the initial designs for the first occurring SCDP run in

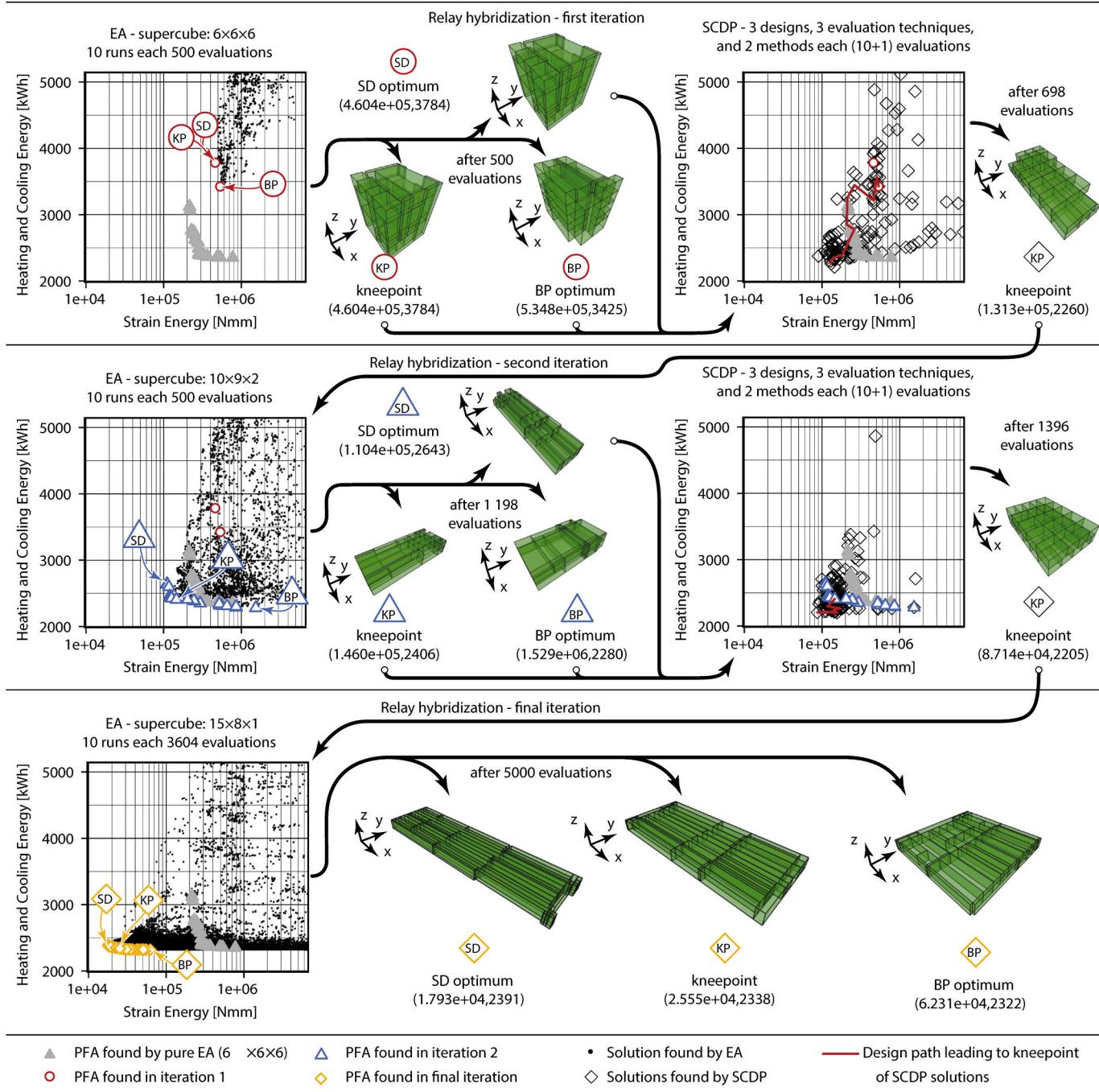


Fig. 9. Results of the relay hybridization.

each method are retrieved from the first occurring exploratory EA, thus in principle—for the first occurring SCDP run—both hybrids are identical. Taking this into account, the relay hybridization is better and should be preferred.

5.4. Comparison

At the hand of the literature review and the results presented in Sections 5.1 to 5.3, a comparison has been made. The comparison is made based on three characteristics: required evaluation budget, sensitivity to initial settings, and optimality.

5.4.1. Required evaluation budget

The EA has been employed with an evaluation budget of 5000 so that

the PFA can converge, and it has been run ten times to reduce its sensitivity to the stochastic initialization and mutation operators, thus in total 50,000 evaluations have been performed. It should be noted that in the context of this comparison it would be more appropriate to compare the SCDP results to the median attainment curve [39], which is a measure of the PFA that the EA is likely to achieve after 1 run. Because the median attainment curve is not available here (see Section 5.1), the total number of performed evaluations by the EA is not taken into consideration for this comparison. Nonetheless, over all SCDP runs, only 198 evaluations have been carried out, which lead to performances that dominate the best PFA that has been found by the EA with a budget of 5000 evaluations. Even when a full convergence of the EA is not required and the evaluation budget is set closer to that of SCDP (e.g. 500), SCDP still achieves better results.

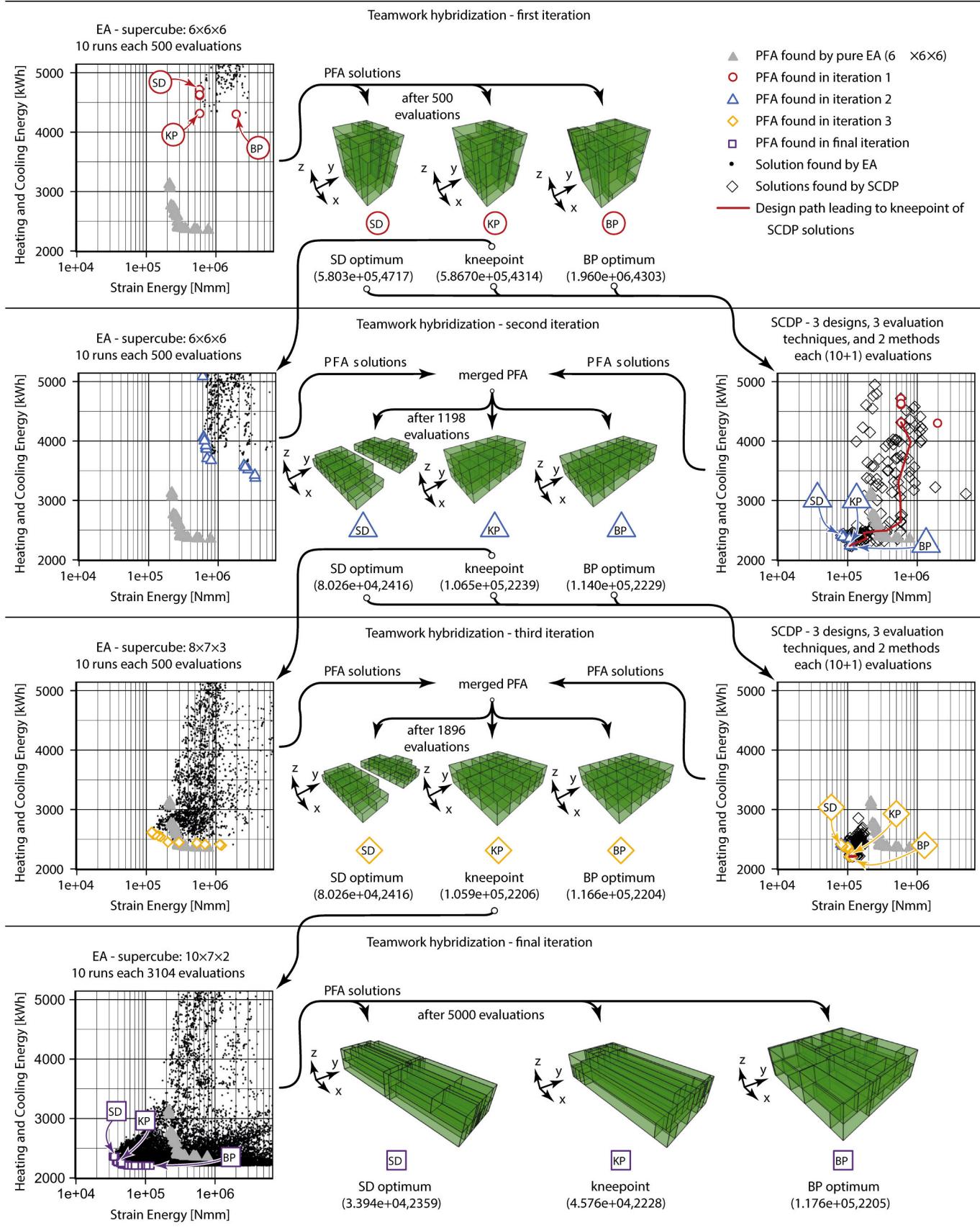


Fig. 10. Results of the teamwork hybridization.

What should be noted on the conclusion drawn from the above observations is that it depends on the definition of the supercube, for instance it is unlikely that the same conclusion could have been drawn if a supercube of two cells in height was used. Via hybridization, the supercube can be adjusted and as such the EA may also find better PFAs. This can be observed from the graphs in Figs. 9 and 10 in which a comparison has been made with the EA results (grey solid triangles) as found in Section 5.1, Fig. 7. When comparing the required evaluation budget, it can be noticed that SCDP finds designs that dominate the designs found by the EA already in the first iteration (after 698 evaluations) of the relay hybrid and in the second iteration (after 1198 evaluations) of the teamwork hybrid. The exploratory EA that follows after the finding of these SCDP solutions yields PFAs that are similar to or even better than the PFA found by the EA demonstration. This occurs in the second iteration of the relay hybrid after 1198 evaluations and in the third iteration of the teamwork hybrid after 1896 evaluations. The hybrid methods can thus find designs with similar performance while using fewer evaluations than the EA.

Similarly, the results of the hybrid methods after the final iterations can be compared to the PFA of the EA (Section 5.1, Fig. 7). It is then observed that the PFAs found by the hybrid methods dominate the PFA found in the EA demonstration. Thus the hybrid methods are able to find designs with better performance than the EA when given the same amount of evaluations, thanks to SCDP.

5.4.2. Sensitivity to initial settings

In the literature review, it is argued that a superstructure may exclude designs from a design search space. This is also the case with the supercube, since the EA could not have found a single storey building spatial design with the used supercube ($6 \times 6 \times 6$ cells and 50 spaces). However, as is observed in the results of the EA in Section 5.1, also the explorability of the tailored SMS-EMOA algorithm of a supercube should be considered when determining the superstructure for an optimization problem. This is because a two storey building has not been found by the EA, even though it can be represented by the used supercube and from the SCDP results in Section 5.2 it is known that two storey design can improve the PFA found by the EA. From the hybrid results in Section 5.3 it can be observed from the visualizations in Figs. 9 and 10 that the EA has also found one and two storey buildings. As such, the hybrid methods are less sensitive to the initial definition of a supercube than the EA alone.

Also SCDP is sensitive to initial settings, as it requires an initial design, which also determines which designs are (im)-possible. Namely, a one storey building (Design 3 in the demonstration, Section 5.2) cannot lead to a multi-storey building with the used SCDP methods when their height is less than 6 (twice the lower bound of the constraint on the height of a space) because the SCDP methods do not scale a design over the z-direction. And, if the initial design consists of spaces with square floor plans, only spaces with a span ratio of 1 : 1 or 1 : 2 can be reached (1 : 4 is not possible because a space will be split across the largest dimension). Whereas an initial design with spaces that have a span ratio of 3 : 5 have shown to lead to better structural designs (Section 5.2, Design 2). Moreover, also the settings and the type of the SCDP method have an influence on the designs that are found. The sensitivities of SCDP with respect to an initial design have also been studied in conjunction with the hybrid methods. With the hybrid method, new design solutions can theoretically gain storeys through the definition of a new supercube, but the number of storeys was still observed to only decrease in the hybrid method. Nevertheless, it has also been observed that the sensitivity of SCDP solutions with respect the span ratios within the initial designs has decreased. This can be explained at the hand of an observed increase in the variation of span ratios within the initial designs for SCDP that are found by the EA. Altogether, it should be noted that the sensitivity that is introduced by the use of SCDP can still be improved with regards to increasing the number of storeys.

5.4.3. Optimality

When comparing the visualized designs that were found by the EA in Section 5.1, Fig. 7 to the visualized designs that have been found by the SCDP methods for Designs 1 and 2 in Section 5.1, (Fig. 8), it is clear that the SCDP methods have yielded the designs with the best performance. However, no clear PFA can be observed from the SCDP performance plots in Fig. 8, whereas a well-distributed PFA is obtained from the EA. A well-distributed PFA can offer insight in the trade-off between the objectives of designs that are similar to the found non-dominated points and it can be used to study the characteristics of optimal solutions. A single non dominated point or a discretely distributed PFA cannot be used to gain such insights. The hybrid methods also yield a well-distributed PFA that can dominate/compete with the solutions found by SCDP. This way, the hybrid methods can find well-distributed PFAs thanks to the EA, and they can find high-quality solutions thanks to SCDP.

6. Discussion

This paper presents three methods for early-stage building spatial design optimization: (I) an evolutionary algorithm, (II) simulations of co-evolutionary design processes, and (III) a hybridization of methods I and II such that the new hybrid inherits their advantages and diminishes their disadvantages. The methods have been applied in a case study, and the results have been analyzed and compared. In this section, critical remarks on the presented methods and work are given.

First, it should be noted that the structural objective is sensitive to the loading that is applied to the structural finite element model by the design grammar. As a consequence, if the ratios between the applied loads change, then one load becomes more dominant and also the solution to the optimal structural design changes. Similarly, when the span distances within a building spatial design are increased, vertical loading becomes more dominant and wind loading less, and vice versa. Additionally, for the structural system, generic material properties and dimensions are used, modifying these will also affect the structural objective. For these reasons it is important that the structural design grammar is used with an a priori notion of the span distances, loads, and structural systems, such that it can be used to qualitatively evaluate a building spatial design, a more in-depth discussion on this topic can be found in Boonstra et al. [21]. Note that similar dependencies may also exist for other settings of either the structural design grammar or the building physics grammar.

Second, as has been mentioned in Section 5.1 and 5.3, the used EA, i.e. the tailored SMS-EMOA algorithm [11], has (great) difficulty with exploring parts of the design search space. It appears that the tailored initialization and mutation operators introduce a bias towards designs that occupy the full supercube. The results are therefore sensitive to the definition of a supercube. Although the hybrid methods improve this issue, it would be interesting to see the effect of hybridization when the EA performs a better exploration of the design search space.

Third, in Section 3.4 only high-level hybridization is selected for the presented work. In a high-level scheme each method is applied in a self-contained approach, whereas in a low-level scheme each method is integrated with the other method(s). The developed high-level methods perform well in the case study (Section 5.3), and thus it would be interesting to also investigate a low-level hybridization scheme. However, a changing supercube size may make a low-level hybridization challenging to implement, because employing an EA designed to work with a superstructure with a changing search space is challenging in itself.

Fourth, in the hybridization schemes presented in Section 3.4 some sub-processes can be investigated in more detail: (a) the selection of solutions from a Pareto front approximation can be substituted with another selection method, e.g. hyper-volume subset selection; (b) the definition of a new supercube is solely based on the knee-point, but it can also be based on a set of solutions that perform well; (c) the initial

population of each EA run is initialized with new designs, and it would be interesting to investigate an initialization that includes SCDP designs into the initial population of an EA. These studies have not been performed in the presented work as it entails an initial study, but they may benefit the performance of the hybrid methods.

Fifth, the sensitivity of the parameters necessary for the hybrid method should be investigated. EA and SCDP already require a significant amount of parameters to be defined and hybridization introduces even more, which may make the configuration of parameters a difficult task. A sensitivity study may offer insight, intuition, and/or recommendations for parameter settings.

Finally, each time the evolutionary algorithm is employed, it is run ten times in order to reduce the sensitivity of the results to the stochastic initialization and mutation operators. Although SCDP is deterministic, the hybrid methods also have a stochastic characteristic because of the use of an EA. It would thus be interesting to develop hybridization schemes that take into account this character, for example by running SCDP on the Pareto front approximation of each EA run, rather than the combined Pareto front approximation over all ten runs.

7. Conclusion and outlook

In this paper three methods for early-stage building spatial design optimization are described and demonstrated: (I) the tailored SMS-EMOA algorithm [11], which is an evolutionary algorithm, (II) two SCDP methods, which simulate co-evolutionary design processes, and (III) a hybridization of methods I and II such that the new hybrid inherits their advantages and diminishes their disadvantages.

The tailored SMS-EMOA algorithm (method I) can find well-distributed Pareto front approximations, which can be used to gain insights in the trade-offs between the objectives and to study the characteristics of optimal solutions. However, disadvantages of the employed evolutionary algorithm are: (i) it requires a large amount of design evaluations (and thus computation time); (ii) the size and complexity of the design search space are limited (i.e. the collection of possible designs); and (iii) it has (great) difficulty exploring all parts of the (already limited) design search space.

Simulations of co-evolutionary design processes (method II) are inspired by design practice in which building engineers apply their creativity, experience, and knowledge in order to find well-performing designs in a large design search space. SCDP methods have been shown to find well-performing solutions with only a small amount of design evaluations from an unrestricted design search space. However, disadvantages of the SCDP methods are: (i) they typically find discretely distributed Pareto front approximations; (ii) they are sensitive to their initial settings; and (iii) they do not give a confidence of convergence, i.e. different settings may yield different results and newly found designs may perform worse than their predecessors but they may still lead to

improved designs.

Hybridization (method III) combines two or more optimization methods into a new one, such that it inherits their advantages and diminishes their disadvantages. Therefore, two hybridization schemes of the tailored SMS-EMOA algorithm and the SCDP methods have been proposed and compared. From a case study it has been concluded that both hybrid methods can: (1) find similar results as the evolutionary algorithm using fewer design evaluations; (2) find better results than the evolutionary algorithm using the same amount of design evaluations; (3) find well-distributed Pareto front approximations; (4) limit the design search space such that it includes optimal designs and can be explored more effectively by the evolutionary algorithm. As such, the advantages of both the SMS-EMOA algorithm and the SCDP methods are inherited, and their disadvantages are diminished. It is thus concluded that—for early-stage building spatial design optimization—the used evolutionary algorithm and the methods that simulate co-evolutionary design processes can successfully be hybridized. While both hybridization approaches had similar final PFAs, relay hybridization has better anytime performance in the proposed setup and is therefore preferred.

Based on the remarks given in the discussion in Section 6, the following recommendations for future work are made: (a) improve the initialization and mutation operators of the tailored SMS-EMOA algorithm, such that it can explore all parts of the (limited) design search space effectively; (b) investigate hybridization schemes that integrate the tailored SMS-EMOA algorithm and the SCDP methods with each other, i.e. so-called low-level hybridization; (c) develop new subprocesses for the presented hybrids, e.g. algorithms to select solutions from a Pareto front approximation, algorithms to define new supercubes based on a set of solutions rather than just one, or include solutions found by SCDP in the initial population of the tailored SMS-EMOA algorithm; (d) investigate the sensitivity of the hybrid methods to the stochastic nature of the tailored SMS-EMOA algorithm and if applicable modify the presented hybridization schemes to reduce that sensitivity; (e) allow building spatial design to have non-orthogonal shapes; (f) include more disciplines, such as lighting.

Declaration of Competing Interest

None.

Acknowledgements

This work is part of the TTW-Open Technology Program with project number 13596, which is (partly) financed by the Netherlands Organization for Scientific Research (NWO). Moreover, the authors would like to acknowledge D.P.H Claessens MSc., Th.Y. de Goede MSc. and T.W. Snel MSc. for their contributions to the toolbox.

Appendix A. Design grammars

This appendix gives a short description of the discipline models that are generated by the so-called design grammars, which are used to evaluate structural and thermal objectives for a given building spatial design. Moreover, the properties of the components with which the models are generated are given (e.g. materials and dimensions), the used solving methods are described, and it is explained how the objectives are computed. However, this section only gives a brief overview, and for a more detailed description of the design grammars, the structural FEM model, and the thermal RC-network, the reader is referred to [20].

A.1. Structural FEM model

For the work in this paper, the structural design grammar has been configured such that it places flat shell components on the boundaries of spaces (walls and floors), thereby ensuring the resulting structural FE model is conformal (i.e. no T-joints between flat shells exist). These flat shell components are given the following properties: thickness $t = 150\text{mm}$; Young's modulus $E = 30\,000\,\text{N}\,\text{mm}^{-2}$, and, Poisson's ratio $\nu = 0.3$. Four wind load cases are defined, one in each azimuthal direction ($\pm x$ and $\pm y$), each flat shell that is located at the boundary of the building spatial design, i.e. is external, is assigned a wind load based on the direction of the wind load case and the direction of the flat shell's normal: $1.0\,\text{kN}\,\text{m}^{-2}$ for pressure loads;

0.8 kN m⁻² for suction; and 0.4 kN m⁻² for shear loads. Moreover, flat shells with vertically oriented normals are assigned a life load in $-z$ -direction, the magnitude of which is 5.0 kN m⁻² for flat shells that are located at the bottom of a space (floor load) and 1.0 kN m⁻² if no space exists above such a flat shell (roof load). The structural model is constrained in its movement in x -, y -, and z -directions at the space edges that are oriented horizontally and that are located at a z -coordinate at or below zero (i.e. $z \leq 0$). Each flat shell component is meshed into 3 by 3 quadrilateral elements describing normal-, shear-, and bending action using 2×2 integration points (Gaussian quadrature). After meshing, the resulting system is solved using a direct solver (the sparse Simplicial-LLT solver of the Eigen C++ library; [44]). The objective for structural design is then calculated by taking the total sum of strain energy [N mm] over all elements in the structural model.

A.2. Thermal RC-network model

The building physics design grammar has been configured such that a construction is placed on the boundaries of spaces (walls and floors) in the building spatial model. All constructions contain a material layer with a thickness of $t = 150\text{mm}$, a specific weight of $\gamma = 2400\text{kgm}^{-3}$, a specific heat capacity of $C = 850\text{JK}^{-1}\text{kg}^{-1}$, and a thermal conduction coefficient of $\lambda = 1.8\text{WK}^{-1}\text{m}^{-1}$, which represents the thermal behavior of concrete. Additionally, constructions that are located at the boundary of a building spatial design, i.e. are external, have an additional layer for insulation, with the following properties: $t = 150\text{mm}$; $\gamma = 60\text{kgm}^{-3}$; $C = 850\text{JK}^{-1}\text{kg}^{-1}$; $\lambda = 0.04\text{WK}^{-1}\text{m}^{-1}$. Accordingly, a resistor-capacitor (RC) network is generated, in which each construction and each space is modeled by a temperature point. The heat capacity of a construction or space is modeled by a grounded capacitor which is connected to the corresponding temperature point. Connections between spaces and constructions are modeled with a resistor that is connecting the two corresponding temperature points. To simulate heating and cooling, an ideal power source with a capacity of 100 W/m³ for both cooling and heating is connected to the temperature point of a space. If the temperature of a space rises above the set-point of 25 °C cooling is activated, and if it drops below the set-point of 20 °C heating is activated. The temperature points of external structure that are located above ground are connected to an independent temperature point with a temperature that is based on real-world measurements obtained at De Bilt in The Netherlands by the Royal Netherlands Meteorological Institute (KNMI) [51]. Two periods of each 3 days (72 h) are simulated, a typical warm period (July 2nd - 4th 1976) and a typical cold period (December 30th 1978 – January 1st 1979). Similarly, the temperature point of constructions that are below ground (e.g. ground floor and basement wall) are connected to a ground temperature point with a constant temperature of 10°C. To ensure all temperature points have a realistic initial temperature at the start of a simulation period, a warm-up period of four days is prepended to each simulation period. The behavior of the RC-network can be expressed in a system of ordinary differential equations, which is solved using a Runge-Kutta stepper (the 5th order Dormand-Prince algorithm from the Odeint C++ library; [1]) using error control: $\epsilon_{abs} = 1 \times 10^{-3}$ and $\epsilon_{rel} = 1 \times 10^{-3}$. The objective for building thermal energy usage is then computed as the sum of heating and cooling power over all simulated periods and all spaces.

A.3. Evolutionary algorithm settings

The settings that have been used for the tailored SMS-EMOA algorithm [11] are described here. The settings for the constraints are as follows: The total floor area of a solution is constrained to match exactly 750 m²; the dimensions of the supercube grid in z -direction are constrained to a value within a range of 3 m to 20 m; and the grid dimensions in x - and y -direction are constrained to values within a range of 0.5 m to 20 m. For the parameters of the tailored SMS-EMOA algorithm, the best performing parameter configuration has been adopted from [11], which are given here in Table B.1. Finally, for the reference point with which the hyper-volume is computed the following point is used ($1 \times 10^{10}, 1 \times 10^{10}$). For a complete and detailed description of the used evolutionary algorithm and its settings the reader is referred to van der Blom et al. [11].

Table B.1

Parameter configuration of the tailored SMS-EMOA algorithm, as found in van der Blom et al. [11].

Symbol	Description	Value
μ	population size	6
<i>MT</i>	mutation type probability	0.4993
<i>ST</i>	step size technique	2
<i>MC</i>	continuous mutation probability	0.4381
<i>IT</i>	initialization technique	2*

* A different initialization technique is used in comparison with the best performing configuration found in van der Blom et al. [11].

A.3.1. Floor area constraint

Instead of the volume constraint that is presented in van der Blom et al. [11], a new constraint is used for the work presented in this paper, namely a floor area constraint. This floor area constraint is enforced by repairing a building spatial design such that it is satisfied, which is explained here in more detail.

Let A_0 be the desired floor area and A_c the current floor area. Then define $\alpha = A_0/A_c$ and scale the width and depth dimensions (x - and y -direction) of the supercube following Eqs. (B.1). If lower or upper bounds are exceeded, set the variable to the bound. Iterate until A_c within one μm^2 of A_0 .

$$\begin{aligned} \forall_i : w_i &= \sqrt{\alpha \cdot w_i} \\ \forall_i : d_i &= \sqrt{\alpha \cdot d_i} \end{aligned} \quad (\text{B.1})$$

References

- [1] K. Ahnert, M. Mulansky, Odeint—solving ordinary differential equations in C++, in: AIP Conference Proceedings, AIP, 2011, pp. 1586–1589, <https://doi.org/10.1063/1.3637934>.
- [2] J.E. Anderson, G. Wulfhorst, W. Lang, Energy analysis of the built environment—a review and outlook, *Renew. Sust. Energ. Rev.* 44 (2015) 149–158, <https://doi.org/10.1016/j.rser.2014.12.027>.
- [3] M.R. Asl, S. Zarrinnmehr, M. Bergin, W. Yan, BPOpt: a framework for bim-based performance optimization, *Energy Build.* 108 (2015) 401–412, <https://doi.org/10.1016/j.enbuild.2015.09.011>.
- [4] T. Bäck, D.B. Fogel, Z. Michalewicz, *Handbook of Evolutionary Computation*, IOP Publishing Ltd, 1997, <https://doi.org/10.1201/9780367802486>.
- [5] S. Bandaru, K. Deb, Temporal innovation: evolution of design principles using multi-objective optimization, in: International Conference on Evolutionary Multi-Criterion Optimization 79–93, Springer, 2015, https://doi.org/10.1007/978-3-319-15934-8_6.
- [6] J.P. Basbagill, F.L. Flager, M. Lepech, A multi-objective feedback approach for evaluating sequential conceptual building design decisions, *Autom. Constr.* 45 (2014) 136–150, <https://doi.org/10.1016/j.autcon.2014.04.015>.
- [7] A.D. Belegundu, S. Rajan, A shape optimization approach based on natural design variables and shape functions, *Comput. Methods Appl. Mech. Eng.* 66 (1988) 87–106, [https://doi.org/10.1016/0045-7825\(88\)90061-8](https://doi.org/10.1016/0045-7825(88)90061-8).
- [8] M. Bendsoe, O. Sigmund, *Topology Optimization. Theory, Methods, and Applications*, second ed., Springer, 2004. ISBN: 978-3-662-05086-6.
- [9] L.C. Bezerra, M. López-Ibáñez, T. Stützle, A large-scale experimental evaluation of high-performing multi-and many-objective evolutionary algorithms, *Evol. Comput.* 26 (2018) 621–656, https://doi.org/10.1162/evco_a.00217.
- [10] E. Biyikli, To, A.C. Proportional topology optimization: a new non-sensitivity method for solving stress constrained and minimum compliance problems and its implementation in matlab, *PLoS One* 10 (2015) 1–23, <https://doi.org/10.1371/journal.pone.0145041>.
- [11] K. van der Blom, S. Boonstra, H. Hofmeyer, T. Bäck, M.T. Emmerich, Configuring advanced evolutionary algorithms for multicriteria building spatial design optimisation, in: 2017 IEEE Congress on Evolutionary Computation (CEC), IEEE, 2017, pp. 1803–1810, <https://doi.org/10.1109/CEC.2017.7969520>.
- [12] K. van der Blom, S. Boonstra, H. Hofmeyer, M.T.M. Emmerich, Multicriteria building spatial design with mixed integer evolutionary algorithms, in: J. Handl, et al. (Eds.), *Parallel Problem Solving from Nature – PPSN XIV*, Springer International Publishing, 2016, pp. 453–462, https://doi.org/10.1007/978-3-319-45823-6_42.
- [13] K. van der Blom, S. Boonstra, H. Hofmeyer, M.T.M. Emmerich, A super-structure based optimisation approach for building spatial designs, in: V. Plevris, M. Papadakis, V. Papadopoulos, G. Stefanou (Eds.), *Congress proceedings of the VII European Congress on Computational Methods in Applied Sciences and Engineering ECCOMAS*, 2016, pp. 3409–3422, <https://doi.org/10.7712/100016.2044.10063>.
- [14] K. van der Blom, S. Boonstra, H. Hofmeyer, M.T.M. Emmerich, Analysing optimisation data for multicriteria building spatial design, in: International Conference on Evolutionary Multi-Criterion Optimization, Springer, 2019, pp. 671–682, https://doi.org/10.1007/978-3-030-12598-1_53.
- [15] K. van der Blom, S. Boonstra, H. Wang, H. Hofmeyer, M.T.M. Emmerich, Evaluating memetic building spatial design optimisation using hypervolume indicator gradient ascent, in: L. Trujillo, O. Schütze, Y. Maldonado, P. Valle (Eds.), *Numerical and Evolutionary Optimization – NEO 2017*, Springer International Publishing, Cham, 2019, ISBN 978-3-319-96104-0, pp. 62–86, https://doi.org/10.1007/978-3-319-96104-0_3.
- [16] K. Bolbotowski, L. He, M. Gilbert, Design of optimum grillages using layout optimization, *Struct. Multidiscip. Optim.* 58 (2018) 851–868, <https://doi.org/10.1007/s00158-018-1930-6>.
- [17] S. Boonstra, K. van der Blom, H. Hofmeyer, J. van den Buijs, M.T. Emmerich, Coupling between a building spatial design optimisation toolbox and Bouwconnect BIM, in: *Advances in Informatics and Computing in Civil and Construction Engineering*, Springer, 2018, pp. 95–102. ISBN: 978-3-030-00220-6.
- [18] S. Boonstra, K. van der Blom, H. Hofmeyer, M.T.M. Emmerich, Combined superstructured and super-structure free optimisation of building spatial designs, in: C. Koch, W. Tizani, J. Ninic (Eds.), *24th EG-ICE International Workshop on Intelligent Computing in Engineering*, University of Nottingham, United Kingdom, 2017, ISBN 978-1-5108-4345-5, pp. 23–34.
- [19] S. Boonstra, K. van der Blom, H. Hofmeyer, M.T.M. Emmerich, Co-evolutionary design processes applied to building spatial design optimization, in: *Proceedings of the 13th World Congress of Structural and Multidisciplinary Optimization, ISSMO*, 2019, pp. 1–6.
- [20] S. Boonstra, K. van der Blom, H. Hofmeyer, M.T.M. Emmerich, J. van Schijndel, P. de Wilde, Toolbox for super-structured and super-structure free multidisciplinary building spatial design optimisation, *Adv. Eng. Inform.* 36 (2018) 86–100, <https://doi.org/10.1016/j.aei.2018.01.003>.
- [21] S. Boonstra, K.V.D. Blom, H. Hofmeyer, M.T.M. Emmerich, Conceptual structural system layouts via design response grammars and evolutionary algorithms, *Autom. Constr.* (2020) 1–20, doi: 10.1016/j.autcon.2019.103009. (accepted 2 nov. 2019).
- [22] S. Boonstra, H. Hofmeyer, Bso Toolbox, 2020, <https://doi.org/10.5281/zenodo.3823893>.
- [23] K. Bringmann, S. Cabello, M.T.M. Emmerich, Maximum volume subset selection for anchored boxes, in: 33rd International Symposium on Computational Geometry (SoCG 2017), Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2017, <https://doi.org/10.4230/LIPIcs.SoCG.2017.22>, pp. 22:1–22:15.
- [24] L. Caldas, Generation of energy-efficient architecture solutions applying gene_arch: An evolution-based generative design system, *Adv. Eng. Inform.* 22 (2008) 59–70, <https://doi.org/10.1016/j.aei.2007.08.012>.
- [25] C.M. Clevenger, J. Haymaker, Metrics to assess design guidance, *Des. Stud.* 32 (2011) 431–456, <https://doi.org/10.1016/j.destud.2011.02.001>.
- [26] C.M. Clevenger, J.R. Haymaker, The value of design strategies applied to energy efficiency, *Smart Sustain. Built Environ.* 1 (2012) 222–240, <https://doi.org/10.1108/20466091211287119>.
- [27] H. Daz, L.F. Alarcón, C. Mourgués, S. Garca, Multidisciplinary design optimization through process integration in the AEC industry: strategies and challenges, *Autom. Constr.* 73 (2017) 102–119, <https://doi.org/10.1016/j.autcon.2016.09.007>.
- [28] W. Dolan, P. Cummings, M. Le Van, Algorithmic efficiency of simulated annealing for heat exchanger network design, *Comput. Chem. Eng.* 14 (1990) 1039–1050, [https://doi.org/10.1016/0098-1354\(90\)85001-Q](https://doi.org/10.1016/0098-1354(90)85001-Q).
- [29] S. Drosté, D. Wiesmann, Metric based evolutionary algorithms, in: European Conference on Genetic Programming, Springer, 2000, pp. 29–43, https://doi.org/10.1007/978-3-540-46239-2_3.
- [30] R.C. Eberhart, Y. Shi, J. Kennedy, *Swarm intelligence*, Elsevier, 2001, ISBN 978-1-55860-595-4.
- [31] T.M. Echenagucia, A. Capozzoli, Y. Cascone, M. Sassone, The early design stage of a building envelope: multi-objective search through heating, cooling and lighting energy performance analysis, *Appl. Energy* 154 (2015) 577–591, <https://doi.org/10.1016/j.apenergy.2015.04.090>.
- [32] ECTP, Challenging and changing Europe's built environment: A vision for a sustainable and competitive construction sector by 2030. Technical Report, European Construction Technology Platform, 2005.
- [33] M. Ehrgott, *Multicriteria Optimization*, Springer-Verlag Berlin Heidelberg, 2005, <https://doi.org/10.1007/3-540-27659-9>.
- [34] M. Emmerich, N. Beume, B. Naujoks, An EMO algorithm using the hypervolume measure as selection criterion, in: C.A. Coello Coello, A. Hernández Aguirre, E. Zitzler (Eds.), *Evolutionary Multi-Criterion Optimization*, Springer, Berlin Heidelberg, Berlin, Heidelberg, 2005, pp. 62–76, https://doi.org/10.1007/978-3-540-31880-4_5.
- [35] M. Emmerich, M. Grötzner, M. Schütz, Design of graph-based evolutionary algorithms: a case study for chemical process networks, *Evol. Comput.* 9 (2001) 329–354, <https://doi.org/10.1162/106365601750406028>.
- [36] M.T.M. Emmerich, A.H. Deutz, A tutorial on multiobjective optimization: fundamentals and evolutionary methods, *Nat. Comput.* 17 (2018) 585–609, <https://doi.org/10.1007/s11047-018-9685-y>.
- [37] S.J. Fenves, H. Rivard, N. Gomez, SEED-Config: a tool for conceptual structural design in a collaborative building design environment, *Artif. Intell. Eng.* 14 (2000) 233–247, [https://doi.org/10.1016/S0954-1810\(00\)00018-2](https://doi.org/10.1016/S0954-1810(00)00018-2).
- [38] U. Flemming, R. Woodbury, Software environment to support early phases in building design (SEED): overview, *J. Archit. Eng.* 1 (1995) 147–152, [https://doi.org/10.1061/\(ASCE\)1076-0431\(1995\)1:4\(147\)](https://doi.org/10.1061/(ASCE)1076-0431(1995)1:4(147)).
- [39] C.M. Fonseca, V.G. da Fonseca, L. Paquete, Exploring the performance of stochastic multiobjective optimisers with the second-order attainment function, in: *International Conference on Evolutionary Multi-Criterion Optimization*, Springer, 2005, pp. 250–264, https://doi.org/10.1007/978-3-540-31880-4_18.
- [40] J.S. Gero, N. D'Cruz, A.D. Radford, Energy in context: a multicriteria model for building design, *Build. Environ.* 18 (1983) 99–107, [https://doi.org/10.1016/0360-1323\(83\)90001-X](https://doi.org/10.1016/0360-1323(83)90001-X).
- [41] P. Geyer, Component-oriented decomposition for multidisciplinary design optimization in building design, *Adv. Eng. Inform.* 23 (2009) 12–31, <https://doi.org/10.1016/j.aei.2008.06.008>.
- [42] P. Geyer, A. Schlueter, Automated metamodel generation for design space exploration and decision-making—a novel method supporting performance-oriented building design and retrofitting, *Appl. Energy* 119 (2014) 537–556, <https://doi.org/10.1016/j.apenergy.2013.12.064>.
- [43] F.W. Glover, G.A. Kochenberger, *Handbook of Metaheuristics* vol. 57, Springer Science & Business Media, 2006. ISBN: 978-0-306-48056-0.
- [44] G. Guennebaud, B. Jacob, et al., Eigen v3: A C++ Linear Algebra Library, URL, <http://eigen.tuxfamily.org>, 2019.
- [45] M. Hamdy, A.T. Nguyen, J.L. Hensen, A performance comparison of multi-objective optimization algorithms for solving nearly-zero-energy-building design problems, *Energy Build.* 121 (2016) 57–71, <https://doi.org/10.1016/j.enbuild.2016.03.035>.
- [46] H. Hofmeyer, J.M. Davila Delgado, Coevolutionary and genetic algorithm based building spatial and structural design, *AI EDAM* 29 (2015) 351–370, <https://doi.org/10.1016/S0890060415000384>.
- [47] C.J. Hopfe, J.L. Hensen, Uncertainty analysis in building performance simulation for design support, *Energy Build.* 43 (2011) 2798–2805, <https://doi.org/10.1016/j.enbuild.2011.06.034>.
- [48] R. Jackson, Optimization of chemical reactors with respect to flow configuration, *J. Optim. Theory Appl.* 2 (1968) 240–259, <https://doi.org/10.1007/BF00937370>.
- [49] R. Kicinger, T. Arciszewski, K. De Jong, Evolutionary computation and structural design: a survey of the state-of-the-art, *Comput. Struct.* 83 (2005) 1943–1978, <https://doi.org/10.1016/j.compstruc.2005.03.002>.
- [50] I.Y. Kim, B.M. Kwak, Design space optimization using a numerical design continuation method, *Int. J. Numer. Methods Eng.* 53 (2002) 1979–2002, <https://doi.org/10.1002/nme.369>.
- [51] KNMI, Measured Weather Data in The Netherlands 53 (8) (2001). <http://www.knmi.nl/nederland-natuur/klimatologie/daggegevens>.
- [52] J.R. Koza, D. Andre, F.H. Bennett III, M.A. Keane, Use of automatically defined functions and architecture-altering operations in automated circuit synthesis with

- genetic programming, in: Proceedings of the First Annual Conference on Genetic Programming, Stanford University MIT Press, Cambridge, MA, 1996, pp. 132–140.
- [53] W.J. Krzanowski, Y.T. Lai, A criterion for determining the number of groups in a data set using sum-of-squares clustering, *Biometrics* 44 (1988) 23–34, <https://doi.org/10.2307/2531893>.
- [54] T. Kuhn, C.M. Fonseca, L. Paquete, S. Ruzika, M.M. Duarte, J.R. Figueira, Hypervolume subset selection in two dimensions: formulations and algorithms, *Evol. Comput.* 24 (2016) 411–425, https://doi.org/10.1162/EVCO_a_00157.
- [55] R.S. Liggett, Automated facilities layout: past, present and future, *Autom. Constr.* 9 (2000) 197–215, [https://doi.org/10.1016/S0926-5805\(99\)00005-9](https://doi.org/10.1016/S0926-5805(99)00005-9).
- [56] V. Machairas, A. Tsangrassoulis, K. Axarli, Algorithms for optimization of building design: a review, *Renew. Sust. Energ. Rev.* 31 (2014) 101–112, <https://doi.org/10.1016/j.rser.2013.11.036>.
- [57] D.J.C. MacKay, *Information Theory, Inference and Learning Algorithms*, Cambridge University Press, 2003, ISBN: 978-0-521-64298-9.
- [58] M.L. Maher, H.H. Tang, Co-evolution as a computational and cognitive model of design, *Res. Eng. Des.* 14 (2003) 47–64, <https://doi.org/10.1007/s00163-002-0016-y>.
- [59] 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, <https://doi.org/10.1016/j.aei.2007.03.003>.
- [60] R. Mora, G. Bitsuamlak, M. Horvat, Integrated life-cycle design of building enclosures, *Build. Environ.* 46 (2011) 1469–1479, <https://doi.org/10.1016/j.buildenv.2011.01.018>.
- [61] C.T. Mueller, J.A. Ochsendorf, Combining structural performance and designer preferences in evolutionary design space exploration, *Autom. Constr.* 52 (2015) 70–82, <https://doi.org/10.1016/j.autcon.2015.02.011>.
- [62] K. Negendahl, T.R. Nielsen, Building energy optimization in the early design stages: a simplified method, *Energy Build.* 105 (2015) 88–99, <https://doi.org/10.1016/j.enbuild.2015.06.087>.
- [63] A.N. Nielsen, R.L. Jensen, T.S. Larsen, S.B. Nissen, Early stage decision support for sustainable building renovation—a review, *Build. Environ.* 103 (2016) 165–181, <https://doi.org/10.1016/j.buildenv.2016.04.009>.
- [64] J. Oh, T. Hong, H. Kim, J. An, K. Jeong, C. Koo, Advanced strategies for net-zero energy building: focused on the early phase and usage phase of a building's life cycle, *Sustainability* 9 (2017) 2272, <https://doi.org/10.3390/su9122272>.
- [65] G.E. Okudan, S. Tauhid, Concept selection methods—a literature review from 1980 to 2008, *Int. J. Design Eng.* 1 (2008) 243–277, <https://doi.org/10.1504/IJDE.2008.023764>.
- [66] M.J. Risbeck, C.T. Maravelias, J.B. Rawlings, R.D. Turney, A mixed-integer linear programming model for real-time cost optimization of building heating, ventilation, and air conditioning equipment, *Energy Build.* 142 (2017) 220–235, <https://doi.org/10.1016/j.enbuild.2017.02.053>.
- [67] M. Ruiz-Montiel, M.V. Belmonte, J. Boned, L. Mandow, E. Millán, A.R. Badillo, J. L. Pérez-de-la Cruz, Layered shape grammars, *Comput. Aided Des.* 56 (2014) 104–119, <https://doi.org/10.1016/j.cad.2014.06.012>.
- [68] A. Schlueter, P. Geyer, Linking BIM and design of experiments to balance architectural and technical design factors for energy performance, *Autom. Constr.* 86 (2018) 33–43, <https://doi.org/10.1016/j.autcon.2017.10.021>.
- [69] Z. Sekulski, Least-weight topology and size optimization of high speed vehicle-passenger catamaran structure by genetic algorithm, *Mar. Struct.* 22 (2009) 691–711, <https://doi.org/10.1016/j.marstruc.2009.06.003>.
- [70] P. Sharafi, B. Samali, H. Ronagh, M. Ghodrat, Automated spatial design of multi-story modular buildings using a unified matrix method, *Autom. Constr.* 82 (2017) 31–42, <https://doi.org/10.1016/j.autcon.2017.06.025>.
- [71] T.W. Snel, *Heuristic Modification Procedures for Building Spatial Design Optimization*, Master's thesis, Eindhoven University of Technology, the Netherlands, 2019.
- [72] H. Song, J. Ghaboussi, T.H. Kwon, Architectural design of apartment buildings using the implicit redundant representation genetic algorithm, *Autom. Constr.* 72 (2016) 166–173, <https://doi.org/10.1016/j.autcon.2016.09.001>.
- [73] B. Steiner, E. Mousavian, F.M. Saradj, M. Wimmer, P. Musialski, Integrated structural–architectural design for interactive planning, *Computer Graphics Forum* 36 (2017) 80–94, <https://doi.org/10.1111/cgf.12996>.
- [74] G. Stiny, *Shape: Talking about Seeing and Doing*, MIT Press, 2006, ISBN 978-0-262-19531-7.
- [75] E.G. Talbi, A taxonomy of hybrid metaheuristics, *J. Heuristics* 8 (2002) 541–564, <https://doi.org/10.1023/A:1016540724870>.
- [76] E. Touloupani, T. Theodosiou, Performance simulation integrated in parametric 3d modeling as a method for early stage design optimization—a review, *Energies* 10 (2017) 637, <https://doi.org/10.3390/en10050637>.
- [77] P. Voll, M. Lampe, G. Wrobel, A. Bardow, Superstructure-free synthesis and optimization of distributed industrial energy supply systems, *Energy* 45 (2012) 424–435, <https://doi.org/10.1016/j.energy.2012.01.041>.
- [78] L. Wang, W. Shen, H. Xie, J. Neelamkavil, A. Pardasani, Collaborative conceptual design—state of the art and future trends, *Comput. Aided Des.* 34 (2002) 981–996, [https://doi.org/10.1016/S0010-4485\(01\)00157-9](https://doi.org/10.1016/S0010-4485(01)00157-9).
- [79] 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, <https://doi.org/10.1007/s12273-011-0052-5>.
- [80] M. Wetter, M. Bonvini, T.S. Nouidui, Equation-based languages—a new paradigm for building energy modeling, simulation and optimization, *Energy Build.* 117 (2016) 290–300, <https://doi.org/10.1016/j.enbuild.2015.10.017>.
- [81] M. Zawidzki, J. Szklarski, Multi-objective optimization of the floor plan of a single story family house considering position and orientation, *Adv. Eng. Softw.* 141 (2020) 1–14, <https://doi.org/10.1016/j.advengsoft.2019.102766>.
- [82] Z. Zhao, J. Wu, Y. Qin, B. Liang, The strong coupled form-finding and optimization algorithm for optimization of reticulated structures, *Adv. Eng. Softw.* 140 (2020) 1–14, <https://doi.org/10.1016/j.advengsoft.2019.102765>.