Multicriteria Building Spatial Design with Mixed Integer Evolutionary Algorithms

Koen van der Blom[1](✉), Sjonnie Boonstra[2], Hèrm Hofmeyer[2], and Michael T.M. Emmerich[1]

[1] Leiden Institute of Advanced Computer Science, Leiden University, Niels Bohrweg 1, 2333 CA Leiden, The Netherlands
{k.van.der.blom,m.t.m.emmerich}@liacs.leidenuniv.nl
[2] Department of the Built Environment, Eindhoven University of Technology, P.O. Box 513, 5600 MB Eindhoven, The Netherlands
{s.boonstra,h.hofmeyer}@tue.nl

Abstract. This paper proposes a first step towards multidisciplinary design of building spatial designs. Two criteria, total surface area (i.e. energy performance) and compliance (i.e. structural performance), are combined in a multicriteria optimisation framework. A new way of representing building spatial designs in a mixed integer parameter space is used within this framework. Two state-of-the-art algorithms, namely NSGA-II and SMS-EMOA, are used and compared to compute Pareto front approximations for problems of different size. Moreover, the paper discusses domain specific search operators, which are compared to generic operators, and techniques to handle constraints within the mutation. The results give first insights into the trade-off between energy and structural performance and the scalability of the approach.

Keywords: Evolutionary algorithms · Super-structure · Mixed integer optimisation · Multicriteria optimisation · Building spatial design · Building structural design · Building physics

1 Introduction

When designing buildings many disciplines have to be taken into account. For example structural design, because a building structure should have optimal strength, stiffness, and stability. Compliance is a specific measure of the stiffness of the building structure and will be subject to investigation in this paper. Another example is building physics, for which in this paper specifically climate control is used as objective, via the minimisation of the building outer surface, being a pre-cursor for future RC-network modelling obtaining minimal energy use for heating and cooling. This is an increasingly important objective due to unpredictable energy prices and climate protection. The built environment is responsible for about 40 % of the total use of energy and materials [1].

J. Handl et al. (Eds.): PPSN XIV 2016, LNCS 9921, pp. 453–462, 2016.
DOI: 10.1007/978-3-319-45823-6_42

Traditionally, energy efficiency and structural design objectives are dealt with in different engineering disciplines, and the same holds for various other objectives (e.g. architectural engineering, construction, etc.). Multidisciplinary optimisation aims to combine different disciplines in order to find building designs that perform well with respect to criteria from various disciplines. It has been used with great success in areas such as automotive and aerospace engineering [2], while in the building design domain its development is still somewhat limited.

This paper advances towards multidisciplinary optimisation of building designs, starting with finding building spatial designs based on criteria from structural design (compliance) and energy efficiency (total surface area). By proposing a multicriteria optimisation approach, the problem of conflicting objectives is discussed. In this case a Pareto front of building designs is computed that can be used in preparation of decision making, to understand design principles that lead to high performance in one discipline or the other discipline, and to find valid compromise solutions.

Traditional algorithms in (evolutionary) multicriteria optimisation, such as SMS-EMOA and NSGA-II, have been formulated for parametric design spaces. For such spaces they have been extensively tested and show a reliable performance. Recently a new super-structure for building spatial design was introduced by the authors [3,4] and here it is used for multicriteria optimisation for the first time. The super-structure encodes building spatial designs by means of a mixed integer representation. By changing discrete variables a large number of alternatives can be encoded. Continuous variables are used to change the dimensioning of these alternatives. Building spatial designs are viewed as configurations consisting of building spaces that do not overlap with each other. To enforce the feasibility of the structural designs generated for the building spatial designs, constraints on the variables are formulated by means of equations, which are checked before evaluation.

Given these preliminaries, this paper will provide the following research contributions: (1) first results on multicriteria optimisation of building spatial designs, including topology choices, (2) discussion of domain specific algorithm design aspects (search operators, constraint handling), and (3) interpretation and discussion of the evolved Pareto fronts in the multidisciplinary building design context. Another aspect discussed in this paper is the scalability of the approach in terms of the size and complexity of the building spatial design.

The remainder of this paper is structured as follows. Section 2 provides a brief summary of building design optimisation and the discipline-specific objectives. Then Sect. 3 discusses multicriteria optimisation techniques. The search space representation, constraints and objective functions are discussed in Sect. 4. Algorithm details are given in Sect. 5. Thereafter, in Sect. 6 numerical results are presented and Sect. 7 discusses these results and provides an outlook.

2 Building Spatial Design

Usually a building is designed by an architect and several engineers. They discuss their progress in project meetings, yet each discipline spends much effort on solving and optimising (discipline specific problems) at their own office. As such, fruitful interaction between disciplines is not guaranteed. This inefficiency of separated disciplines in the built environment gained acknowledgement [5], which gave rise to tools that allow more direct collaboration between engineers. One such tool is building information modelling (BIM) [6]. Through the modelling of data from various disciplines BIM allows information to be shared between engineers working on different building design aspects. Since choices made during the early stages of a design naturally propagate to the later stages, tighter collaboration by employing such tools may avoid one discipline disproportionally affecting performances in other disciplines.

An overview of optimisation tools in the built environment is provided by Palonen et al. [7]. Such tools generally parametrise components of the building design to enable the optimisation. Often these tools are limited to variation of the design through alteration of component variables, adding new components is rarely possible. Advances are made though, for example in the work by Hofmeyer and Davila Delgado [8], which focuses on optimisation via the simulation of a co-evolutionary preliminary building design process. Another interesting work is that of Hopfe et al. [9] where the significance of design variables on the building physics performance is predicted.

3 Multidisciplinary and Multicriteria Optimisation

Recently it has been recognised [5] that in order to help design teams consisting of experts from different disciplines in finding solutions, objectives and simulations from different disciplines have to be considered in concert. Multicriteria optimisation can be an important method in this context, as it allows to deal with conflicting objectives and can effectively support decision making.

In general, a multicriteria optimisation problem (MOP) is defined by a set of objective functions $f_i : X \to \mathbb{R}$, $i = 1, \dots, m$ to be minimised (or maximised) for some search space X. Moreover, constraint functions $g_j(\boldsymbol{x})$ are usually considered, the value of which must be kept within a prescribed range.

For two feasible solutions $\boldsymbol{x}$ and $\boldsymbol{x}'$, it is said that $\boldsymbol{x}$ (Pareto) dominates $\boldsymbol{x}'$, if and only if $\forall i = 1, \dots, m$: $f_i(\boldsymbol{x}) \leq f_i(\boldsymbol{x}')$ and there exists $j = 1, \dots, m : f_j(\boldsymbol{x}) < f_j(\boldsymbol{x}')$. The efficient set X_E is the subset of X consisting of points that are not dominated by any point in X. The set $\{(f_1(\boldsymbol{x}), \dots, f_m(\boldsymbol{x}))^T | \boldsymbol{x} \in X_E\} \subset \mathbb{R}^m$ is called the Pareto front (PF) of the MOP (given it exists). The PF provides valuable information about the space of all relevant solutions and their trade-offs. This paper aims to compute the PF for the real world problem of building spatial design and discuss the trade-offs between discipline specific objectives.

Recently, various powerful black box optimisation algorithms have been proposed for approximating Pareto fronts. Many of these belong to the class of evolutionary multicriterion optimisation, which use selection and variation (stochastic mutation, recombination) to steer a population of search points close to the Pareto front. The selection operator needs to take into account Pareto dominance, but diversity maintenance is also important in order to guarantee that all parts of the Pareto front are covered.

Two state-of-the-art evolutionary multicriterion optimisation algorithms, namely NSGA-II [10] and SMS-EMOA [11] are used as basic strategies in this paper. These algorithms will be instantiated for a domain specific search space.

4 Formal Problem Specification

4.1 Search Space Representation

The supercube representation, recently proposed by the authors [3,4], serves to represent the design space by means of continuous and discrete variables. The goal of the supercube representation was to formulate building design optimisation as a mixed integer nonlinear programming (MINLP) problem, an approach that in other domains is typically referred to as super-structure-based optimisation. Essentially, discrete variables encode the topology of spaces in the building spatial design and continuous variables determine the dimensioning of the spaces.

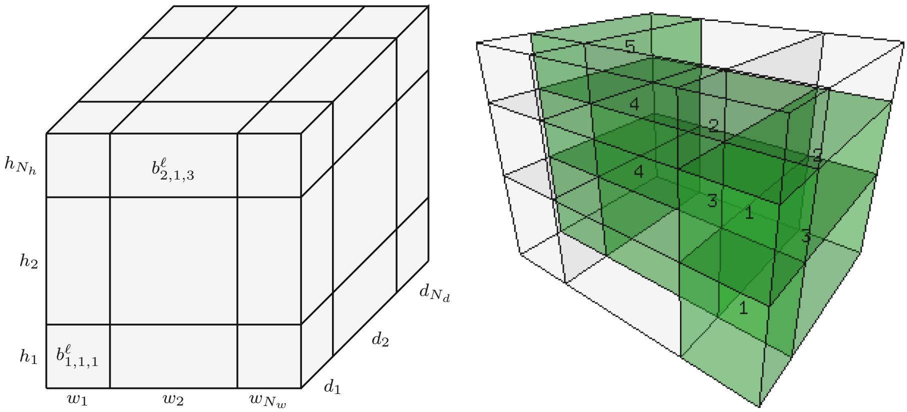

Fig. 1. Supercube grid representation (left) and building spatial design (right).

Building spatial designs consisting of N_{spaces} spaces are encoded in a cuboid (3D rectangle) grid of $N_w \times N_d \times N_h$ cells, these variables respectively refer to the number of cells in width, depth and height directions. In turn those same directions employ the indices $i \in \{1, \ldots, N_w\}$, $j \in \{1, \ldots, N_d\}$ and $k \in \{1, \ldots, N_k\}$, to determine their dimensioning with the variables w_i, d_j and h_k. Finally each cell may be turned on or off as being part of a space $\ell \in \{1, \ldots, N_{spaces}\}$ by the binary variable $b^{\ell}_{i,j,k}$. This is referred to as the supercube representation, Fig. 1 shows an example of a supercube and a derived building spatial design.

4.2 Topology Constraints

Four topology constraints are considered to disallow configurations of the supercube that are infeasible from an engineering point of view. All of these constraints can be described in mathematical form with just sums and products as presented by the authors in [3,4]. Textual explanations of the constraints and an example of the mathematical notation follow.

No Overlap ensures each cell is active for at most one space which can be defined mathematically with Eq. 1. Spaces should have a **Cuboid Shape**. This can be checked in two steps. Firstly it is ensured that for every space active cells appear at the same indices in all distinct rows, columns and beams. Secondly it is checked there are no gaps between the active cells of a space. **Vertical Gaps** between spaces, like archways and cantilevered parts, are disallowed in order to facilitate the check to determine whether a building stands on the ground by simple procedures. Finally a **Constant Number of Spaces** is enforced by making sure every space consists of at least one cell.

$$\forall_{i,j,k} : \sum_{\ell=1}^{N_{spaces}} b^{\ell}_{i,j,k} \leq 1 \tag{1}$$

4.3 Objective Functions

Energy performance is measured as the total outside surface area of the building spatial design, excluding the floor surface of the ground level. In the future, a RC-network model is planned to find heating and cooling energy per space.

For structural performance a black box simulator is used (meaning standard MINLP solvers cannot be used for optimisation) with the following settings. First the building spatial design is provided with a structural design via a so-called structural grammar. The grammar used here adds four concrete walls and a concrete roof (a slab) to every space, both with a thickness t=150 mm. Young's modulus of the concrete is set to E=30000 N/mm^2 and Poisson's ratio to $v = 0.3$. Live loads of 1.8 kN/m^2 are then applied to each slab, and wind loads from eight directions (N, NW, W, etc.) are applied to the building spatial design (with a pressure of 1.0 kN/m^2, a suction of 0.5 kN/m^2 and a shear of 0.4 kN/m^2) and transferred to the structural design. Using a finite element analysis (FEM), the compliance over all loads is calculated. For more details, see [8].

5 Algorithm Design

5.1 Volume Repair

A fixed volume V_0 for the building spatial design will be maintained during optimisation because otherwise objectives could possibly be optimised largely by taking extreme values for the continuous variables. The volume is taken as in Eq. 2 below. To exclude inactive cells $b_{i,j,k}$ is found by: $b_{i,j,k} = \sum_{\ell=1}^{N_{spaces}} b^{\ell}_{i,j,k}$, note that Eq. 1 needs to hold.

$$\sum_{i=1}^{N_w} \sum_{j=1}^{N_d} \sum_{k=1}^{N_h} b_{i,j,k} w_i d_j h_k = V_0 \tag{2}$$

When the volume of a new individual is not within a 1 % deviation of V_0 it is repaired by scaling the continuous variables. After scaling, continuous variables exceeding the lower bound are set to the lower bound. Variables exceeding the upper bound are multiplied by 0.95 until their value is within the bound. Naturally changes to variable values will also change the volume, therefore the process is repeated until the bound checks succeed without changes to the variables. Using the desired volume and the current volume V_c a factor $\alpha = V_0/V_c$ may be computed. Multiplying the dimensions of the supercube with the cubic root of α results in V_0. As such the scaling function is described by Eq. 3.

$$\forall_i : w_i = \sqrt[3]{\alpha w_i} \qquad \forall_j : d_j = \sqrt[3]{\alpha d_j} \qquad \forall_k : h_k = \sqrt[3]{\alpha h_k} \tag{3}$$

5.2 Optimisation and Constraint Checking

NSGA-II and SMS-EMOA are used with typical settings in the experiments below. In most cases they use the same settings and operators; otherwise it is indicated. A lower bound $lb = 3$ and upper bound $ub = 19.8$ are used for the continuous variables. Selection strategies are $(20 + 20)$ for NSGA-II and $(50 + 1)$ for SMS-EMOA. For the ease of notation $N_{cells} := N_w \times N_d \times N_h$ is defined. Binary variables have a probability of $1/N_{cells}$ to be initialised to one, or zero otherwise. Continuous variables are set to a value from $lb + (ub - lb) \times U$, where U is drawn uniformly at random from $]0, 1]$. Moreover, a fixed step size $0.05 \times (ub - lb)$ is used for the continuous variables. Following the initialisation the volume of the parent population is repaired as described in the previous subsection with a desired volume $V_0 = 4^3 \times N_{cells}$. Each individual is evaluated as follows. If any constraint is violated a penalty value pen is returned based on the number of violations CV such that $pen = 999,999,999 + CV - 1$. Here CV is an integer from one to five to indicate the number of violations. The five constraints relate to the four previously described constraints. The two parts of the cuboid shape constraint are counted separately. The objective functions are only evaluated when no constraints are violated. An evaluation budget of 2500 is used in the experiments. Note that constraint checks are not considered as evaluations here.

Each offspring is created by applying crossover and mutation. For crossover a parent $P1$ is selected uniformly at random from the population. A second parent $P2$ is then selected uniformly at random with a probability of 0.5, otherwise $P2 = P1$. Parents are either recombined with a probability of 0.5, or copied to the different children $C1$ and $C2$. Each binary variable is recombined as $C1 = P1$ and $C2 = P2$ with probability 0.5, or as $C1 = P2$ and $C2 = P1$ otherwise. Simulated binary crossover is applied to the continuous variables. When a variable exceeds a bound it is set to lb or ub as applicable. Finally either of the children is selected with probability 0.5. Mutation is applied with a probability of $1/N_{dims}$, where $N_{dims} := N_{cells} \times N_{spaces} + N_w + N_d + N_h$ is the total number of variables. Binary variables are mutated by bit flips. Polynomial mutation is applied to continuous variables above the lower bound, variables exactly at the boundary are reinitialised (as previously described). Following mutation variables exceeding their bounds are set to their appropriate boundary values. The volume of the produced offspring is repaired as previously described. NSGA-II then applies non dominated/crowding distance sorting to the population of size $\mu + \lambda$ before selecting the first μ individuals for the next parent population. SMS-EMOA selects based on the hypervolume contribution (reference point $(1.1e9, 1.1e9)$).

5.3 Smart Mutation

The general mutation and recombination operators used in NSGA-II and SMS-EMOA have difficulties navigating heavily constrained objective landscapes, such as considered here. A smart mutation operator is proposed which only produces mutants that do not violate the problem specific constraints. Since the algorithms have similar performance only SMS-EMOA is considered with smart mutation.

The smart mutation method works by extending or reducing spaces by either adding or removing a surface of cells. This is done by selecting one of the following faces of the space to make either an outward or an inward move: left, right, top, bottom, front or back. All moves are applied to all cells along the selected face of a space, such that the space remains cuboid when adding and removing cells. These moves are of size one, meaning that the width, depth or height (depending on the selected face) of a space grows or shrinks by a single cell. Whenever an outward move adds a cell to a space A that is already part of a space B the cell is set to inactive for space B. From all mutation steps that do not result in a constraint violation one is chosen uniformly at random.

A new offspring individual is then created as follows. A parent is selected uniformly at random. Smart mutation is applied with a 0.25 probability, otherwise a continuous variable that is relevant to at least one active cell is selected uniformly at random and mutated by polynomial mutation. No crossover is used.

Initialisation of binary variables is changed to ensure the initial population consists solely of valid individuals. For every space a non-fully occupied pillar is selected uniformly at random from the supercube and the first cell from the bottom that does not belong to any previously initialised space is set active for this space. To increase diversity in the initial population twenty smart mutations are applied to the initial individuals of single cell spaces.

Penalty values are no longer used since all offspring are now guaranteed to be valid. The remaining procedures are the same as in Subsect. 5.2.

6 Numerical Results

Problem configurations are denoted by four numbers. The first three indicate the dimensions of the supercube and the last indicates the number of building spaces that are considered. For example 2225 indicates a problem with a $2 \times 2 \times 2$ supercube and five spaces. Every experiment averages over five runs using average Pareto fronts (median attainment curves [12]). Tests were done for 222 and 333 configurations both with one, three and five spaces.

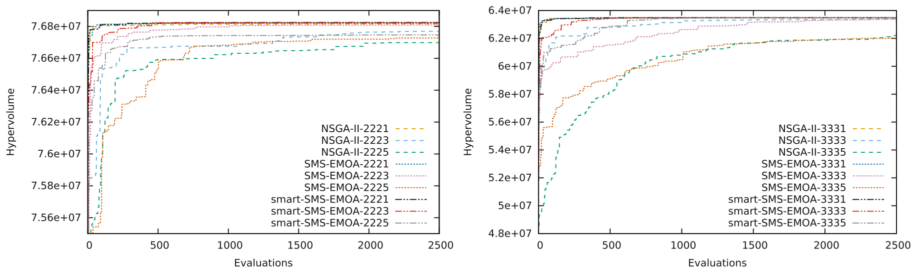

Fig. 2. Average hypervolume growth over five runs, reference point (35000, 2500), for one, three and five spaces in 222 (left) and 333 (right) configurations.

The various problem configurations show a quick convergence to a relatively stable hypervolume (taken with $log(1 + compliance), surface\ area)$ in Fig. 2, with more complicated configurations naturally taking move evaluations before stabilising. NSGA-II and SMS-EMOA produce similar attainment curves as may be observed in Fig. 3. This indicates the considered process works and results in a Pareto front approximation. The standard deviations of the hypervolume at the final generation are relatively small for most problem configurations and do not change the numerical result. Only for the 3335 configuration large deviations occur for the generic methods, but even their highest hypervolume solutions do not outperform the smallest hypervolume found by the method with smart mutation. A one sided Wilcoxon test between NSGA-II and SMS-EMOA results in $W = -1$, indicating there is no significant difference. Moreover, applying the one sided Wilcoxon test between either of those methods and smart SMS-EMOA results in $W = 15$, indicating the method with smart mutation is better with a statistical significance of 0.05.

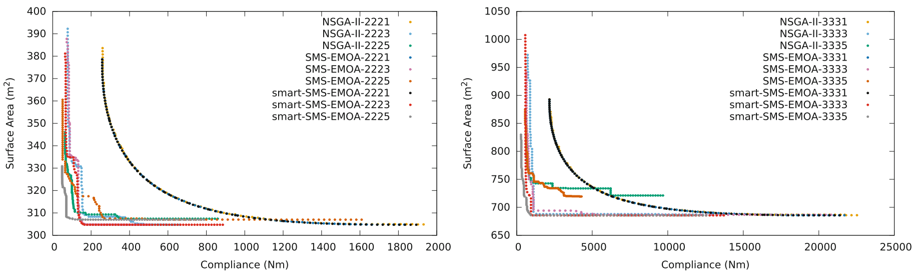

Fig. 3. Median attainment curves from five runs for one, three and five spaces in a 222 configuration (left) and a 333 configuration (right).

Smart SMS-EMOA produces similar results to the other two approaches for single space problems as can be observed in Fig. 2. For the problems with three spaces the method with smart mutation improves over the other two by a decent margin, and for five spaces it is clearly better both in terms of convergence speed and the final solution. The same behaviour can be observed in Fig. 3, where

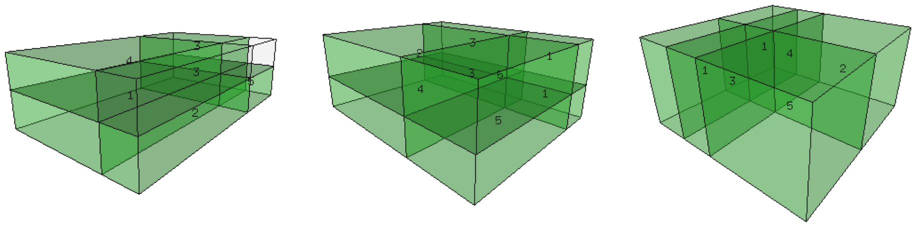

Fig. 4. Best spatial designs found with smart SMS-EMOA for the 3335 configuration. Minimal compliance (left), knee point (center) and minimal surface area (right).

differences in performance become more pronounced with larger problem sizes. Clearly, smart mutation produces a better Pareto front approximation.

Figure 4 shows the best found spatial designs in terms of each objective as well as a compromise solution at the knee point of the median attainment curve. As can be expected the optimal spatial design in terms of minimal surface area has a cuboid shape. The knee point solution is largely similar, but has a slightly lower structure and as a result is stretched in both width and depth to maintain the volume. Finally the minimal compliance solution has an L-shaped and elongated structure. The lower structure can be explained since it results in less strain on the structural elements, reducing the compliance.

Table 1. Average runtime over five runs, rounded to the closest whole minute.

Problem configuration	2221	2223	2225	3331	3333	3335
CPU time (minutes)	42	342	888	42	620	1008

Table 1 shows the CPU time used with smart mutation. The other methods performed similarly because the compliance computations used by far the most CPU time. Each experiment used a single core of an i7-3770 CPU @ 3.40 GHz processor and with 16 GiB DIMM DDR3 Synchronous 1600 MHz memory.

7 Discussion

Multicriteria optimisation algorithms for a building spatial design have been developed and tested for moderate size problems. The problem has been formulated as a mixed integer program. Moreover, the problem is characterised by a large number of constraints and a specific constraint handling mutation operator has been proposed. Pareto front approximations have been obtained. They always have a convex shape which makes it possible to find compromise solutions in knee points. The results show that smart mutations can be beneficial for exploring larger and more dense regions. However, in order to scale up the problem size further research in this direction is needed, including recombination operators. Moreover, surrogate modelling may allow for a more efficient exploration of the objective landscape. Finally, while statistical significant improvement was shown when using the method with smart mutation, parameter tuning should be applied in future work to compare the methods with their optimal parameter settings.

Acknowledgments. The authors gratefully acknowledge the financing of this project by the Dutch STW via project 13596 (Excellent Buildings via Forefront MDO, Lowest Energy Use, Optimal Spatial and Structural Performance).

References

1. European Commission: Challenging and Changing Europes Built Environment: A Vision for a Sustainable and Competetive Construction Sector By 2030. European Construction Technology Platform (2005)
2. Liao, X., Li, Q., Yang, X., Zhang, W., Li, W.: Multiobjective optimization for crash safety design of vehicles using stepwise regression model. Struct. Multi. Optim. **35**(6), 561–569 (2008)
3. van der Blom, K., Boonstra, S., Hofmeyer, H., Emmerich, M.T.M.: A superstructure based optimisation approach for building spatial designs. In: ECCOMAS 2016, 5–10 June, Greece (2016, accepted)
4. Boonstra, S., van der Blom, K., Hofmeyer, H., Amor, R., Emmerich, M.T.M.: Super-structure and super-structure free design search space representations for a building spatial design in multi-disciplinary building optimisation. In: EG-ICE 2016, 29 June–1 July, Poland (2016, accepted)
5. Martins, J.R., Lambe, A.B.: Multidisciplinary design optimization: a survey of architectures. AIAA J. **51**(9), 2049–2075 (2013)
6. Eastman, C., Eastman, C.M., Teicholz, P., Sacks, R.: BIM Handbook: A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers and Contractors. Wiley, Hoboken (2011)
7. Palonen, M., Hamdy, M., Hasan, A.: MOBO a new software for multi-objective building performance optimization. In: Wurtz, E. (ed.) Proceedings of the 13th Internationcal Conference of the IBPSA, pp. 2567–2574. IBPSA c/o Miller-Thompson, Toronto (2013)
8. Hofmeyer, H., Davila Delgado, J.M.: Coevolutionary and genetic algorithm based building spatial and structural design. Artif. Intell. Eng. Des. Anal. Manuf. **29**(04), 351–370 (2015)
9. Hopfe, C.J., Emmerich, M.T.M., Marijt, R., Hensen, J.L.M.: Robust multi-criteria design optimisation in building design. In: Proceedings of Building Simulation and Optimization, Loughborough, UK, pp. 19–26. IBPSA, England (2012)
10. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. **6**(2), 182–197 (2002)
11. Emmerich, M.T.M., Beume, N., Naujoks, B.: An EMO algorithm using the hypervolume measure as selection criterion. In: Coello Coello, C.A., Hernández Aguirre, A., Zitzler, E. (eds.) EMO 2005. LNCS, vol. 3410, pp. 62–76. Springer, Heidelberg (2005)
12. Fonseca, C.M., da Fonseca, V.G., Paquete, L.: Exploring the performance of stochastic multiobjective optimisers with the second-order attainment function. In: Coello Coello, C.A., Hernández Aguirre, A., Zitzler, E. (eds.) EMO 2005. LNCS, vol. 3410, pp. 250–264. Springer, Heidelberg (2005)