

Optimizing solar access and density in Tel Aviv: Benchmarking multi-objective optimization algorithms

Thomas Wortmann¹ and Jonathan Natanian^{2,3}

¹Department of Architecture and Urban Planning, Institute for Computational Design and Construction, University of Stuttgart, Stuttgart, Germany

²Department of Architecture, Chair of Building Technology and Climate Responsive Design, Technical University of Munich, Munich, Germany

³Faculty of Architecture and Town Planning, Technion – Israel Institute of Technology, Haifa, Israel

thomas.wortmann@icd.uni-stuttgart.de

Abstract. This paper explores the trade-off between redeveloping an urban site with higher density and maintaining solar access for the surrounding context in the hot and dry climate of Tel Aviv. Such trade-offs are important for future urban development in the Middle East, where densification is a demographic and environmental need. We explore this trade-off with multi-objective optimization (MOO). Specifically, we benchmark seven MOO algorithms on two test problems with different, parametric typologies: courtyard and high-rise. For both problems, we aim to maximize Floor Area Ratio and the simulation-based Context Exposure Index, a novel metric based on the Israeli green building code. The high-rise emerges as the better performing typology, and HypE, SPEA2, and RBFMOpt as the most efficient and robust MOO algorithms.

1. Introduction

The design of more sustainable, high density urban neighborhoods is an urgent need due to increasing urbanization and environmental challenges, first and foremost global warming. Computational methods can support the design of such neighborhoods, for example through parametric generation, performance simulation, and optimization [1]. Multi-objective optimization (MOO) is an especially promising method, since urban designers must often balance conflicting objectives, for example, density and solar access. Density is an important factor for the future sustainability of built environments in the Middle East and beyond [2]. In terms of solar access, higher density can increase self-shading, thermal comfort, and energy efficiency and reduce glare. On the other hand, it will reduce daylight availability, solar generation potential and passive heating.

But there are not many studies of the applications of MOO to sustainable urban design, and algorithmic performance comparisons, i.e., benchmarks, are almost non-existent [3]. This paper benchmarks seven MOO algorithms—including RBFMOpt, a novel, machine-learning-related algorithm—on two related, simulation-based urban design problems. The problems explore the redevelopment of an urban area in Tel Aviv, Israel in terms of Floor Area Ratio (FAR) and solar access for the surrounding context, but with different typologies: Courtyard and high-rise.

We choose a courtyard typology due to its good performance in previous studies [3,4], and the high-rise typology due to its prevalence for new developments in Tel Aviv. This paper thus illuminates the trade-off between urban density and solar access in hot and dry climates relative to these two typologies, and the relative performance of MOO algorithms on simulation-based urban design problems. Urban designers likely use Pareto fronts from a single MOO algorithm and run to select a non-dominated



solution for further development and/or to understand the tradeoff between FAR and CEI: As such, it is important that the resulting fronts are as accurate as possible [5].

2. Methodology

This section introduces the test problems, MOO algorithms, and benchmark methodology.

2.1. Test Problems

The test problems consider the redevelopment of a site in the greater metropolitan area of Tel Aviv, Israel. The site is around 5,500 square meters large and currently occupied by buildings with FAR 2.6. For the first problem, we consider a courtyard typology and for the second a high-rise typology (see Figure 1). The courtyard's footprint is 72.6 by 52.8 meters, the courtyard itself 39.6 by 19.8 meters, and the high-rise's footprint 33 by 33 meters. The typologies' building mass is varied by parametrizing the heights of their corner points while their footprints remain fixed. For the courtyard, these heights can vary between 3.3 and 47.2 meters, and for the high-rise between 3.3 and 129 meters. These bounds limit FAR to around 7.5 for both typologies. Varying the heights of the four corners independently results in a wide range of building geometries, and, from the perspective of MOO, two problems with four continuous variables each. (For more details on the courtyard's parametric model, see [6].)

For both problems, we want to maximize FAR and the Context Exposure Index (CEI). We want to maximize FAR to achieve a higher urban density, which is necessary to accommodate Israel's projected urban growth [7] and associated with gains in energy efficiency that are comparable to gains from improvements in building technology [2].

CEI is a new metric proposed in another paper submitted to CISBAT 2021 [6]. This metric records the simulated solar exposure of the rooftops, southern facades, and outdoor surfaces of the surrounding context on the 21st of December that is affected by the maximum building mass (for the courtyard and high-rise, respectively). The metric computes an average of this solar exposure relative to a set of thresholds from the Israeli green building code. Maximizing this metric aims to maintain adequate solar access for the surrounding context. In short, the optimization problems consist of maximizing density on the chosen site while maintaining solar exposure for the surrounding context. On an Intel Core i7-6700@3.4 GHz, a single iteration—generating the geometry in Grasshopper and simulating the solar exposure with Ladybug 1.1.0's "Direct Sun Hours" component [8]—takes an average of 3.7 seconds for the courtyard and 5.5 seconds for the high-rise typology.

In a preliminary study of the courtyard typology, where we ran the NSGA-II algorithm for 1,500 iterations, we also included maximizing the New Exposure Index (NEI), which applies the CEI calculation to the new building mass, as an objective. But we found no tradeoff between CEI and NEI, and only a mild tradeoff between FAR and NEI, which is confirmed by the values in Figure 1. In other words, one can achieve adequate solar exposure for both the surrounding context and the new building mass. One also can achieve high density and adequate solar exposure for the new building mass. The most important tradeoff, and the appropriate application of MOO [5], thus lies between CEI and FAR.

2.2. Multi-objective Optimization Algorithms

We explore the tradeoff between CEI and FAR with seven MOO algorithms. These algorithms include the evolutionary HypE, MOEA/D, NSGA-II and SPEA2, the swarm-based NSPSO, the ant colony-based MHACO, and the model-based RBFMOpt. HypE, MOEA/D, NSGA-II and SPEA2 employ similar genetic operators such as mutation and crossover, but differ in terms of their selection operators: HypE uses the hypervolume metric (discussed in section 2.3. NSGA-II uses non-dominated sorting, i.e., sorting solutions in terms of the Pareto rank they achieve, and a crowding metric [9]. SPEA2 also uses non-dominated sorting, but in combination with a fitness value that considers how many other solutions a solution dominates [10]. NSPSO combines non-dominated sorting with particle swarm optimization [11], and MHACO combines the hypervolume metric with ant colony optimization [12].

RBFMOpt extends the model-based RBFMOpt algorithm [13] with a decomposition approach similar to MOEA/D. RBFMOpt is a machine-learning-related algorithm that refines a surrogate model of the fitness landscape during the optimization process. This surrogate model speeds-up RBFMOpt's

optimization process, which makes it especially suitable for expensive (i.e., time-consuming) optimization problems [14]. RBFMOpt restarts RBFMOpt during the optimization process, but with different Tchebycheff scalarizations. (Each scalarization corresponds to a point on the Pareto front.) But, in contrast to MOEA/D, RBFMOpt reuses results from previous scalarizations by feeding them back into the surrogate model. In this way, the surrogate model improves its accuracy during the optimization process and thus supports the optimization process more effectively.

On an urban optimization problem similar to the ones presented here—maximizing FAR and monthly load match for a residential district in Shanghai, China—RBFMOpt outperformed HypE and NSGA-II [4]. On a sustainable design problem of minimizing total energy and maximizing continuous daylight autonomy for the windows of a sports hall in Tallinn, Estonia, RBFMOpt and HypE outperformed MOEA/D, NSGA-II, NSPSO, SPEA2, and MHACO [15]. This benchmark contributes to a generalization of these findings.

Except HypE, SPEA2, and RBFMOpt, we use implementations from Pygmo 2.16.1 [16], a library for parallel optimization in Python. For RBFMOpt, we use an implementation developed by the first author. These implementations are available via Opossum [17], and HypE and SPEA2 via Octopus [18].

2.3. Benchmark

For both problems, we run each algorithm for 1,000 function evaluations (i.e., simulations). This function evaluation budget might appear small, given the higher difficulty of MOO problems. Nevertheless, this budget allowed the better performing algorithms to converge (Figure 2 and Figure 5). With a population size of 40, this budget results in 25 generations for the population-based algorithms (all expect RBFMOpt). Except for this population size, which we choose to allow a reasonable number of generations, we use default settings for each algorithm’s hyperparameters. To account for the algorithms’ use of randomness, we run each algorithm ten times.

We evaluate the algorithms’ performance based on the hypervolume they achieved for each iteration and run. The hypervolume measures the volume (for two objectives, the area) covered by a Pareto front relative to a reference point. Hypervolume is the most relevant performance measure for MOO and “possesses good mathematical properties, ... can capture dominance properties and distribution and does not require the knowledge of the Pareto front” [19]. We chose the nadir, i.e., a point whose coordinates correspond to the worst-known objective value for each objective, as the reference point for each problem. We use Pygmo for the hypervolume calculations.

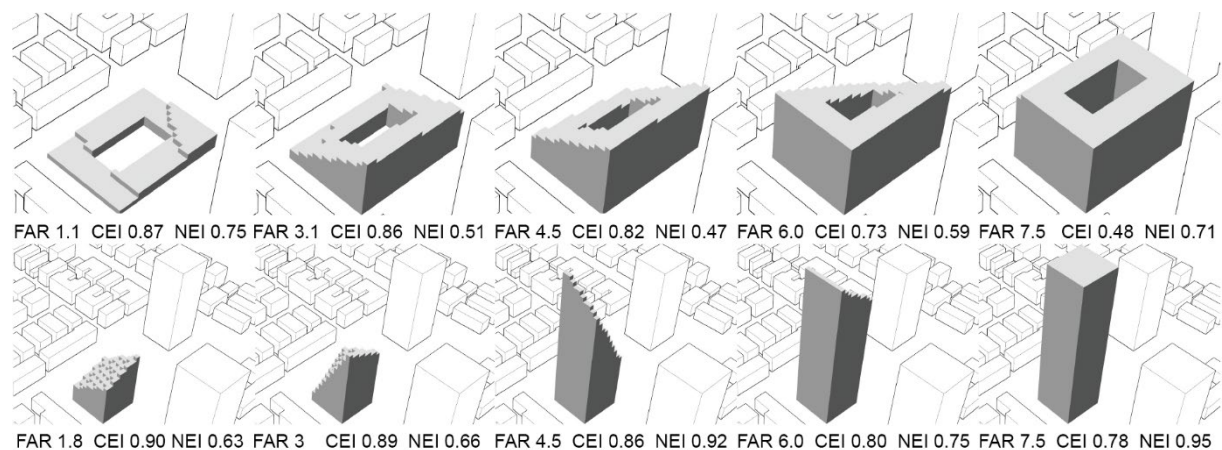


Figure 1 Examples of the courtyard and high-rise buildings from the best-known Pareto fronts, with FAR, CEI, and NEI.

3. Results

This section presents benchmark results for the courtyard and high-rise problems. It discusses the normalized hypervolumes achieved by the algorithms in terms of speed of convergence (i.e., how fast did the algorithms’ respective Pareto fronts improve?) and robustness (i.e., how much did the algorithms’ performance vary on repeated runs?). We assess convergence in terms of the median

hypervolume per iteration and algorithm, and robustness in terms of the resulting hypervolume per run and algorithm. Convergence and robustness matter to urban designers that require fast and reliable optimization results. We illustrate these results with the median Pareto front for each algorithm.

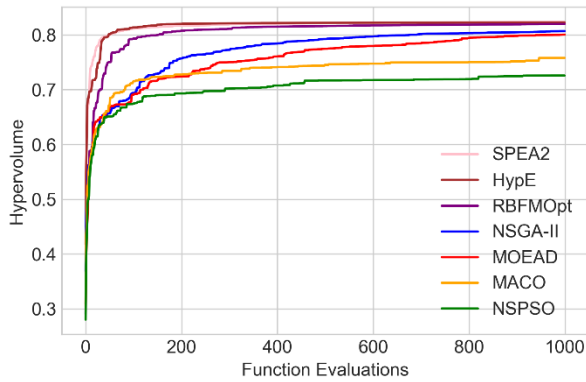


Figure 2 Convergence Courtyard

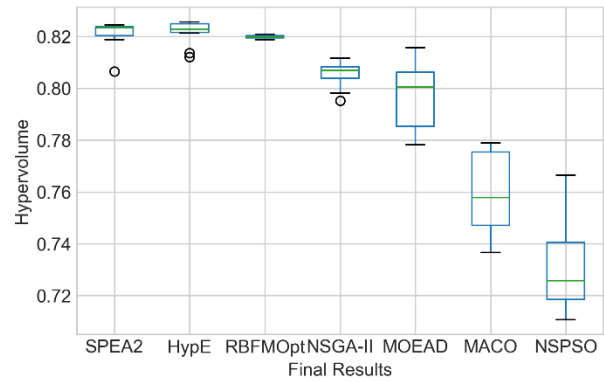


Figure 3 Robustness Courtyard

3.1. Courtyard

On the courtyard problem, HypE, SPEA2, and RBFMOpt converge about equally quickly and achieve similarly good results (Figure 2). NSGA-II and MOEAD perform slightly worse, but ahead of NSPSO and MACO. In terms of robustness, RBFMOpt exhibits the smallest variance, followed by HypE, NSGA-II, and SPEA2, which all result in at least one worse-performing outlier (Figure 3). The large variances of MOEAD, MACO, and NSPSO limits their suitability when designers can perform only a single run, such as in fast-paced conceptual design phases.

The algorithms' median Pareto fronts illustrates the findings on their relative performance (Figure 4). For each Pareto front, the circles represent non-dominated solutions, i.e., solutions where an increase in FAR results in a decrease in CEI, and vice-versa. The “best-known” front represents the non-dominated solutions from all algorithms and runs, and thus is the most accurate. The closer a MOO algorithm comes to this “best-known” front in a single run, the better it is. HypE, SPEA2, and RBFMOpt in most places closely track the best-known Pareto front (which is calculated from all benchmark results), NSGA-II somewhat less closely, and MOEAD, NSPSO, and MACO open relatively large gaps with the best-known front.

The best-known Pareto front indicates little conflict between FAR and CEI up until FAR 4, when CEI starts to decline from about 0.85 to 0.7 for FAR 6.5. For large FAR values around 7.5, one can achieve CEI around 0.55. In short, one can significantly increase the existing density of FAR 2.6 without impacting the solar access of the urban context. But a larger increase reduces this solar access.

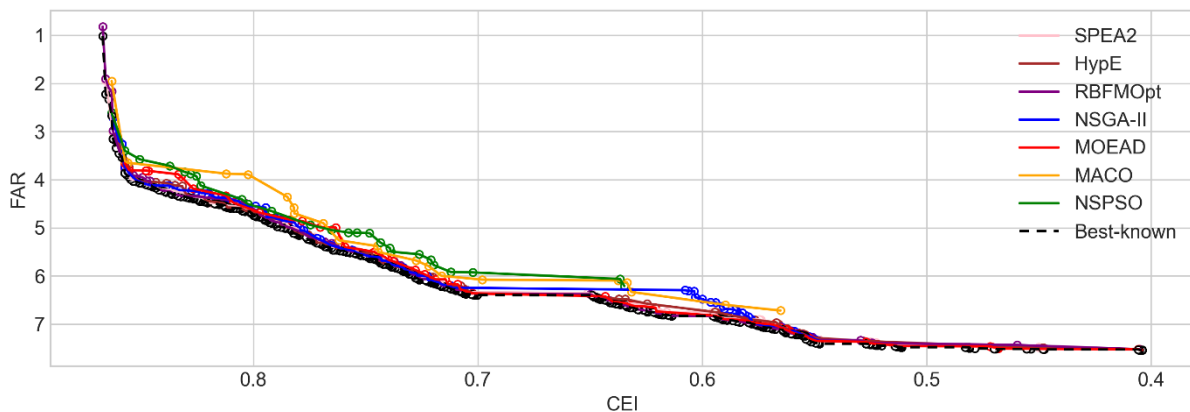


Figure 4 Median Pareto Fronts Courtyard

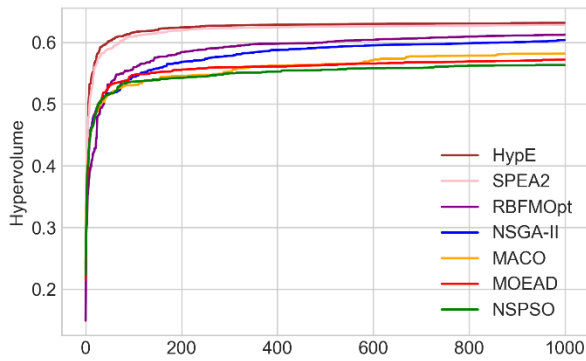


Figure 5 Convergence High-rise

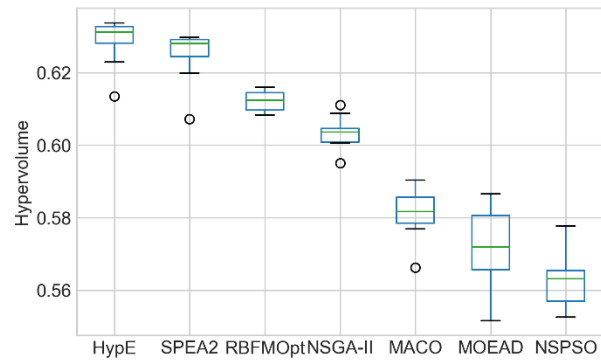


Figure 6 Robustness High-rise

3.2. Highrise

One the high-rise problem, HypE and SPEA2 converge fastest, followed by RBFMOpt and NSGA-II. MACO, MOEAD and NSPSO converge slowest (Figure 5). In terms of robustness, RBFMOpt again exhibits the smallest variance, followed by NSGA-II, HypE, SPEA2, and MACO, which all result in at least one worse-performing outlier (Figure 6). NSPSO and MOEAD again vary the most.

The median fronts of HypE, SPEA2, and RBFMOpt closely track the best-known front (Figure 7). The front of NSGA-II is somewhat offset. MACO, MOEAD and NSPSO open large gaps.

While the algorithms' performance on the two problems is similar, the trade-off between FAR and CEI differs significantly between the courtyard and high-rise typologies: With the high-rise typology, CEI falls more gradually relative to FAR, and still reaches 0.78 for FAR around 7.5. In short, the high-rise typology preserves CEI much better than the courtyard typology, especially for higher FARs.

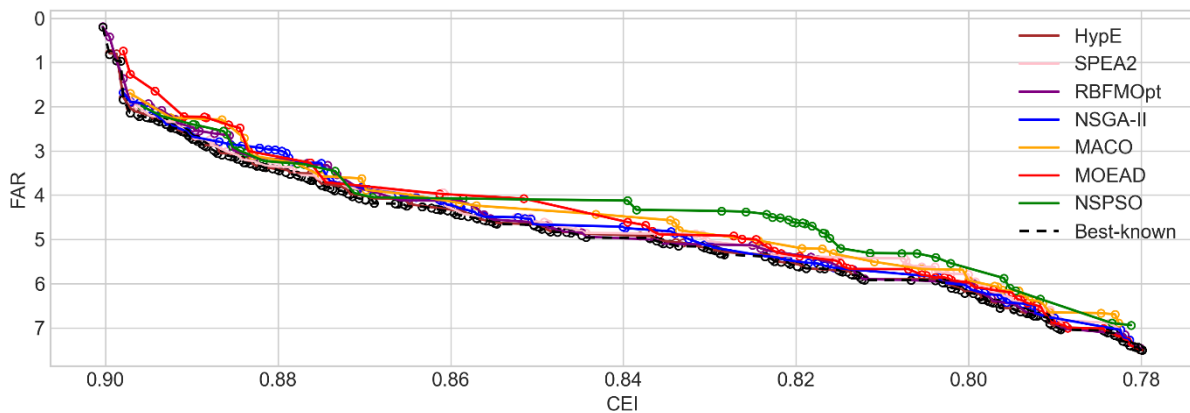


Figure 7 Median Pareto Fronts High-rise

4. Discussion and Conclusion

On both problems, HypE, SPEA2, and RBFMOpt exhibit the fastest convergence. RBFMOpt is the most robust algorithm, but lags behind SPEA2's and RBFMOpt's convergence for the high-rise problem. NSGA-II achieves middling performance on both problems. MACO, MOEAD, and NSPSO are the bottom performers, except for the middling convergence (but large variance) of MOEAD on the courtyard problem. These results help designers and researchers to make better decisions in terms of which MOO algorithms to use on similar problems. But generalizing these findings requires more extensive benchmarks of simulation-based MOO problems. Nevertheless, given the strong performance of RBFMOpt in [4], and of RBFMOpt and HypE in [15], RBFMOpt, HypE, and—possibly—SPEA2 are especially promising for further investigation.

In terms of the trade-off between density and solar access in hot and dry climates, the results appear conclusive: High-rise typologies are more suitable for than courtyard ones (Figure 1). Contrastingly, for Shanghai's hot summer cold winter climate, courtyard districts perform better than high-rise districts in

terms of maximizing FAR and monthly load match, using building-integrated photovoltaics [4]. Future studies that more holistically consider the relationships between metrics such as embodied carbon, energy use, energy generation, daylight quality, and outdoor comfort will further illuminate the relative performance of courtyard and high-rise typologies in different climates.

Acknowledgment

The Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) partially supported this paper under Germany's Excellence Strategy – EXC 2120/1 – 390831618.

References

- [1] Shi Z, Fonseca JA, Schlueter A. A review of simulation-based urban form generation and optimization for energy-driven urban design. *Build Environ* 2017;121:119–29.
- [2] Güneralp B, Zhou Y, Ürge-Vorsatz D, Gupta M, et al. Global scenarios of urban density and its impacts on building energy use through 2050. *Proc Natl Acad Sci* 2017;114:8945–50.
- [3] Natanian J, Wortmann T. Simplified evaluation metrics for generative energy-driven urban design: A morphological study of residential blocks in Tel Aviv. *Energy Build* 2021;240:110916.
- [4] Wortmann T, Natanian J. Multi-objective Optimization for Zero-Energy Urban Design in China: A Benchmark. *Proc. Symp. Simul. Archit. Urban Des.*, Vienna, AU: SCS; 2020, p. 203–10.
- [5] Wortmann T, Fischer T. Does Architectural Design Optimization Require Multiple Objectives? A critical analysis. *Proc. 25nd CAADRIA Conf.*, Bangkok, TH: CAADRIA; 2020.
- [6] Natanian J, De Luca F, Wortmann T, Capeluto G. The Solar Block Generator: an additive parametric method for solar driven urban block design in hot climates. *CISBAT*, 2021.
- [7] Hason S, Kotock O, Drukman D, Roter D. *Israel 2048: Strategic Thinking towards Regional Planning and Development*. IL: Israel Planning Authority and Hebrew University of Israel; 2016.
- [8] Sadeghipour RM, Pak M. Ladybug: a parametric environmental plugin for grasshopper to help designers create an environmentally-conscious design. *13th Int. IBPSA Conf.*, Lyon, FR: 2013.
- [9] Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans Evol Comput* 2002;6:182–97.
- [10] Zitzler E, Laumanns M, Thiele L. *SPEA2: Improving the strength Pareto evolutionary algorithm*. Zurich, CH: Computer Engineering and Networks Laboratory (TIK) Department of Electrical Engineering Swiss Federal Institute of Technology (ETH); 2001.
- [11] Li X. A Non-dominated Sorting Particle Swarm Optimizer for Multiobjective Optimization. In: Cantú-Paz E, Foster JA, Deb K, Davis LD, Roy R, O'Reilly U-M, et al., editors. *Genet. Evol. Comput.* — GECCO 2003, Berlin, Heidelberg: Springer; 2003, p. 37–48.
- [12] Acciarini G, Izzo D, Mooij E. MHACO: a Multi-Objective Hypervolume-Based Ant Colony Optimizer for Space Trajectory Optimization. *2020 IEEE Congr. Evol. Comput. CEC*, 2020.
- [13] Costa A, Nannicini G. RBFOpt: an open-source library for black-box optimization with costly function evaluations. *Math Program Comput* 2018;10:597–629.
- [14] Wortmann T. Genetic Evolution vs. Function Approximation: Benchmarking Algorithms for Architectural Design Optimization. *J Comput Des Eng* 2019;6:414–28.
- [15] De Luca F, Wortmann T. Multi-Objective Optimization for Daylight Retrofit. *Anthropol. Archit. Fabr. Cogn. Age - Proc. 38th ECAADe Conf.*, vol. 1, Berlin, DE: eCAADe; 2020, p. 57–66.
- [16] Biscani F, Izzo D. A parallel global multiobjective framework for optimization: pagmo. *J Open Source Softw* 2020;5:2338.
- [17] Wortmann T. Opossum—Introducing and Evaluating a Model-based Optimization Tool for Grasshopper. In: Janssen P, Loh P, Raonic A, Schnabel MA, editors. *Proc. 22nd CAADRIA Conf.*, Hong Kong, CN: CAADRIA; 2017, p. 283–92.
- [18] Vierlinger R. *Octopus*. Vienna, AUT: Bollinger+Grohmann Engineers; 2012.
- [19] Audet C, Bignon J, Cartier D, Le Digabel S, Salomon L. Performance indicators in multiobjective optimization. *Eur J Oper Res* 2021;292:397–422.