

Article

Genetic and Swarm Algorithms for Optimizing the Control of Building HVAC Systems Using Real Data: A Comparative Study

Alberto Garces-Jimenez ^{1,*}, Jose-Manuel Gomez-Pulido ², Nuria Gallego-Salvador ²
and Alvaro-Jose Garcia-Tejedor ^{1,†}

¹ Centro de Innovación Experimental del Conocimiento (CEIEC), Universidad Francisco de Vitoria (UFV), Carretera Pozuelo-Majadahonda, Km. 1.8, Pozuelo, 28223 Madrid, Spain; a.gtejedor@ceiec.es

² Departamento de Ciencias de la Computación, Universidad de Alcalá (UAH), Carretera Madrid-Barcelona, Km. 33.6. Alcalá de Henares, 28805 Madrid, Spain; jose.gomez@uah.es (J.-M.G.-P.); mnuria.gallego@edu.uah.es (N.G.-S.)

* Correspondence: alberto.garces@ufv.es or albertogarces0@gmail.com; Tel.: +34-696-018-282

† These authors equally contributed.

Citation: Garces-Jimenez, A.; Gomez-Pulido, J.-M.; Gallego-Salvador, N.; Garcia-Tejedor, A.-J. Genetic and Swarm Algorithms for Optimizing the Control of Building HVAC Systems Using Real Data: A Comparative Study. *Mathematics* **2021**, *9*, 2181. <https://doi.org/10.3390/math9182181>

Academic Editor: Freddy Gabbay

Received: 14 July 2021

Accepted: 3 September 2021

Published: 7 September 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).

Abstract: Buildings consume a considerable amount of electrical energy, the Heating, Ventilation, and Air Conditioning (HVAC) system being the most demanding. Saving energy and maintaining comfort still challenge scientists as they conflict. The control of HVAC systems can be improved by modeling their behavior, which is nonlinear, complex, and dynamic and works in uncertain contexts. Scientific literature shows that Soft Computing techniques require fewer computing resources but at the expense of some controlled accuracy loss. Metaheuristics-search-based algorithms show positive results, although further research will be necessary to resolve new challenging multi-objective optimization problems. This article compares the performance of selected genetic and swarm-intelligence-based algorithms with the aim of discerning their capabilities in the field of smart buildings. MOGA, NSGA-II/III, OMOPSO, SMPSO, and Random Search, as benchmarking, are compared in hypervolume, generational distance, ϵ -indicator, and execution time. Real data from the Building Management System of Teatro Real de Madrid have been used to train a data model used for the multiple objective calculations. The novelty brought by the analysis of the different proposed dynamic optimization algorithms in the transient time of an HVAC system also includes the addition, to the conventional optimization objectives of comfort and energy efficiency, of the coefficient of performance, and of the rate of change in ambient temperature, aiming to extend the equipment lifecycle and minimize the overshooting effect when passing to the steady state. The optimization works impressively well in energy savings, although the results must be balanced with other real considerations, such as realistic constraints on chillers' operational capacity. The intuitive visualization of the performance of the two families of algorithms in a real multi-HVAC system increases the novelty of this proposal.

Keywords: multi-objective optimization; genetic algorithms; evolutionary computation; swarm intelligence; Heating, Ventilation and Air Conditioning (HVAC); metaheuristics search; bio-inspired algorithms; smart building; soft computing

1. Introduction

Global energy consumption has been growing at 1.4% annually over the last 10 years [1], and 94% of it is produced with combustion [2]. Greenhouse gas emissions produce adverse effects on the environment and society and cannot be completely replaced. Buildings consume on average 40% of the electrical energy in European Union cities and 32%

in world cities [3], where the Heating, Ventilation, and Air Conditioning (HVAC) system requires 32.7% of the supplied electricity and up to 40.3% in public buildings [4].

Advanced control systems improve energy management by adapting fast to unforeseen events or predicting system behavior. There are some examples of this, such as the application of neural networks with genetic algorithms in building management systems, reaching savings of 27% [5]. Other studies improve the cost by 19.7%, adding an optimization module in the ambient controller [6]. Some researchers have proven that it is possible to save 30% on cold days, embedding a machine-learning-based MPC controller [7]. On the other hand, the faster the controller reaches the goals, the better the energy efficiency is obtained; for example, Adaptive LAMDA-PI (Learning Algorithm for Multivariable Data Analysis—Proportional Integral) controllers improve the Integral Absolute Error (IAE) of the response time by above 140% compared with conventional PI and Fuzzy-PI controllers [8]. Optimization works are embedded in different tasks or problems of the HVAC systems in both design and operations. They are used to adjust Proportional, Integral, and Derivative (PID) controllers to improve the logic of Model-Predicting Controllers (MPCs) or to enhance the supervision tasks in the Building Management Systems (BMSs) or Multi-Agent Controllers (MACs) [9]. There is a significant interest in embedding advanced, Artificial Intelligence (AI)-based control architectures in the BMS [10] that provide acceptable results in uncertain contexts and complex systems, while allowing the adoption of multi-objective optimization policies. There are two visible advanced control strategies: (1) predicting the system behavior with machine-learning-based simulations to obtain the optimal sequence of instructions or (2) adapting the system parameters in case of context perturbances, so that it quickly returns to the zero-error state, such as with fuzzy logic control. Artificial Intelligence (AI), together with other technologies, such as Big Data, Internet of Things (IoT), or Cloud Computing, enhances the ubiquity, accessibility, mobility, knowledge extraction, and autonomy for the new software tasks. The traditional multi-objective problem in operations is to improve the energy efficiency and maintain comfort for the users, i.e., the ideal temperature, humidity, or Indoor Environmental Quality (IEQ) that mutually conflicts. Comfort, health, or maintenance add other objectives to the optimization problem, such as the CO₂ concentration, reducing the efficiency of the optimization with fewer objectives [11].

Zadeh conceptually grouped under the umbrella of “soft computing” (SC) technologies that overperformed traditional deterministic approaches [12], at the expense of losing accuracy and generalization. Thus, SC is tolerant to imprecision and uncertain approximation and today are widely used for complex problems where moderate precision and generalization capability are acceptable, given their high-resolution speed. SC covers three main fields: (1) Machine Learning (ML), (2) metaheuristics-based optimization, and (3) Fuzzy Logic (FL) for decision-making. Metaheuristics-based optimization [11] offers good tradeoffs between consumed resources and accuracy for achieving global goals but brings challenges to face, such as algorithm convergence, stability, parameter tuning, a mathematical framework, benchmarking, generalization, and performance assessment [13]. SC also offers fitness estimation for optimization with data-based models that require fewer computer resources [14].

Digital transformation and the social trend towards standardization allow for sharing the functionality among different fields, requiring testing their approaches and convenience for specific applications. This conceptual ‘liquidity’ brings new challenges for optimization, such as the smart city, smart districts, and smart building, which leads to the scaling of the control and supervision capabilities to upper layers (e.g., ISA 95 and IEC 62264 L2), but constraining the lower layers. More conflicting objectives, such as the Coefficient of Performance (COP), allow for the monitoring of subtle equipment degradations, achieving considerable savings in the life cycle of the installations [15]. The system management is susceptible to becoming autonomous with the self-optimization organic function.

Thus, society, while aiming to enhance people's wealth and comfort, is forced to save energy and reduce costs. Multi-objective optimization strategies can be applied at several levels in building systems, especially HVAC, that can run at bare equipment control, at subsystems management, or at a superuser level integrating systems, buildings, blocks, or districts. In this scenario, there is a greenfield to explore, including among others autonomous building management architectures that automatically adapt their decisions to contextual changes and continuously improve with the experience. The proposed study demonstrates different multi-objective optimization techniques under this scenario that include conventional conflicting goals of comfort, observable with the ambient temperature, and energy-saving, quantifiable with the subsystems consumption, and add two new objectives: (1) the maximization of the absolute value of COP, allowing for optimal performance in saving energy and, at the same time, an enhancement of the lifecycle of the equipment, something rarely explored in operations before [16]; (2) the minimization of the rate of change in ambient temperature, which allows the system to enter into a steady-state mode since startup at nearly critical damping. The possibility for the system to automatically select the most appropriate algorithm is also proposed for the next research outcome. Although it was expected that the addition of conflicting objectives could reduce the efficiency of the optimization, the results show evidence of a wide field to be explored.

This comparative study shows the pros and cons of using different population-based multi-objective optimization algorithms for an HVAC control system. Current practices limit operation to ensure the comfort of building inhabitants dodging other objectives such as energy savings. The study will cover (1) Swarm Intelligence (SI) algorithms and (2) Genetic Algorithms (GAs) and will use real data from the HVAC system of Teatro Real de Madrid (Opera House). The individuals in the decision space are mapped in the objective space with cost functions empirically obtained with ML's Random Forest Regressors (RFRs) to assess their dominance. The RFRs have been trained with a selection of data obtained from a historic database kindly provided by the Board of Teatro Real. The selected GAs are the Multi-Objective Genetic Algorithm (MOGA) and the Non-dominated Sorting Genetic Algorithm version 2 and 3 (NSGA-II and NSGA-III), and the selected SI-based algorithms are Optimized Multi-objective Particle Swarm Optimization (OMOPSO) and Speed-constrained Multi-objective Particle Swarm Optimization (SMPSO). In the experiment, the Strength Pareto Evolutionary Algorithm Version 2 (SPEA2) was discarded, as its execution time was excessive compared to the others. Random Search (RS) results are exhibited as a reference point.

The paper is organized as follows: In Related Work, the authors bring to light significant research related to this study. Materials and Methods explain how the experiment was built and the metrics for comparing the algorithms. The Results section visualizes and discusses the outcomes. Finally, the Conclusions section compares the obtained results with other studies, outlines the novelty, and proposes possible future research lines for this work.

2. Related Work

2.1. Towards a Clear Ontology

It is common for recent literature about SC and multi-objective optimization to take for granted the approach followed in this work, due to the absence of effective classification and, therefore, the formation of an adequate body of knowledge. Although it is beyond the scope of this study, it is prudent to indicate some examples of confusing terms and try to position them.

Non-preference multi-objective optimization, i.e., those finishing with a set of non-dominant solutions, is sometimes classified as a subset of 'a posteriori' decision-making, and sometimes they are synonyms. It is often associated with multimodal optimization, although only the latter also includes local search. It is also difficult to differentiate Evolutionary Computation from GAs. While sharing a similar process, a GA includes mating

and crossover to improve the search. For some articles, they are synonyms and come grouped either as evolutionary or genetic. They are sometimes considered a subset of different approaches, such as bio-inspired algorithms.

Particle Swarm Optimization (PSO) can be classified itself [17] or together with GAs [18] under Multi-Objective Evolutionary Algorithms (MOEA). MOGA is sometimes considered a separated GA [19] or the family of multi-objective GAs [20]. MOEA and MOGA may include the whole metaheuristics-based search family or only those based on the population approach.

The new algorithms based on the observation of nature can be named bio-inspired, bio-search heuristics, or metaphor-based metaheuristics, among others, exchanging their different inspirations, be they biological, chemical, or physical. GAs or SI-based algorithms can be found included in the bio-inspired family, excluding the evolutionary algorithms [19].

With regard to the optimization performance, it is possible to mislead concepts such as ‘convergence’ that could mean either to end the search at any point (lumps) or end it in true global optima. The diversity feature sometimes indicates the uniformity in the distribution of the solutions, how they spread, or both.

The classification by Ahma et al. [9] and Oliva et al. [21] with Zadeč’s original SC definition [12] supports the position of this study. At the top, algorithms are split up into stochastic techniques and intelligent agents (deterministic). Stochastic techniques then split into population-based and single individual algorithms (trajectory metaheuristics) that include Simulated Annealing (SA) and Tabu Search (TS). Population-based algorithms then split into SI and evolutionary algorithms. This study compares GAs (part of evolutionary algorithms) and SI-based particle swarms, PSO.

2.2. Research Interest

According to Wang, G. [22], at an early stage, optimization methods diversified in different fields of study: (1) linear or nonlinear programming, (2) constraints, (3) single- or multi-objective optimization, and (4) dynamic programming. The first generation introduced the iteration and gradients. The second generation brought the metaheuristics-based search for global multi-objectives that reduced computing resources and allowed for parallel computation. Soft Computing (SC) AI approaches support surrogate-based or metamodel generation, replacing computer-aided simulation software with ML models. The next-generation links and hybridizes the above approaches.

Nabaei et al. [23] provide a good reference for research interests in SI algorithms and GAs over time. GAs have been interesting since before 2000 with a peak from 2006 to 2010. PSO algorithms started to become comparable in 2006 and 2010, but there were much fewer articles published than there were regarding GAs, half of them spanning from 2011 to 2018. Another comprehensive study by Shaikh et al. [11] illustrates the research interest for the optimization in building HVAC systems in which GA articles are 24% of the total and MOGA represent 3%. PSO is present in 5%, and MOPSO in 7%. Scheduling Optimization, Hooke and Jeeves, and Linear Quadratic shares range between 3% and 6%.

Optimization can be used for designing systems or in real-time operations [24]. A GA is used for both design and operations, and so is NSGA-II, but only in a third of the articles reviewed. There are more articles about PSO in operations than in design, but a Differential Evolutionary (DE) algorithm is only used in designing, and the number of articles about the combination of these algorithms is similar to the number of articles related to NSGA-II.

2.3. Genetic and Swarm Intelligence Outcomes

Algorithms based on metaheuristics are good options for characterizing the behavior of complex, dynamic, and nonlinear systems [25].

A GA puts together a set of individuals (chromosomes) ‘coded’ with genes (variables), marking them with fitness functions. It then uses a selection strategy to obtain a new

population ready for the next iteration. Mutation and crossover operators regulate the speed and variety of chromosome changes in the GA. While the crossover ‘exploits’ the search, the mutation widens the explored space. One key point is the adjustment of the parameters to the specific problem. The mutation operator can generate solutions with polynomial or uniform probability distributions. The non-uniform probability prevents the population from decaying in the early stages of the evolution by generating distant solutions with a random probability. Simulated Binary Crossover (SBX) generates offspring from two parents attending to their probability distributions.

GAs discover the optimal set in three different ways [26]. (1) The first approach is known as Pareto-based dominance with a two-level ranking scheme, one to obtain the dominance and diversity assessment and the other, containing such metrics as the total nondominated vectors generation, the hypervolume, the generational distance or spacing, and the error rate [27], to determine the convergence to local or global minima [28]. NSGA-II and SPEA2 make use of these principles. (2) The second approach uses unary or binary indicators to check their performance, for example, with the coefficient of determination, the R2-like S-Metric Selection Evolutionary Multi-Objective Optimization Algorithm (SMS-EMOA) that maximizes the hypervolume (HV). (3) The third approach is based on decomposition that splits the overall problem into smaller problems for the search. There is not a common procedure for these algorithms. Splitting up complicated Pareto Fronts (PFs) to apply a local search and Tchebycheff’s scalarization is one of these methods, as well as the Multi-Objective Evolutionary Algorithm based on decomposition (MOEA/D) and NSGA-III.

The advantages of GAs are that they (1) have simple fitness arrangement schemes; (2) do not need derivatives or gradients; (3) are relatively robust; (4) are easy to parallelize. However, although they require less information about the problem, (1) designing an objective function, (2) getting a representation, and (3) adjusting the operators can be a difficult task. In addition, they are computationally expensive compared with others. NSGA and NSGA-II perform niching, decide deterministically the tournaments, and avoid chaotic perturbations of the population composition with updated fitness sharing. However, the niching function is too complex and scales poorly as the number of objectives increases [24].

SI-based optimization is also population-based, where its individuals are bio-inspired on natural ecosystem metaphors, such as ants, bees, or particles [29]. Swarm algorithms still generate some skepticism because of the mentioned metaphoric ornaments describing their operators [30].

In the case of PSO, the particles move around in the decision space with simple mathematical equations that yield their position and velocity. Each particle’s best-known local position and velocity determine its movement towards the optimum. PSO (1) is easy to adjust; (2) can be implemented and provide fast speed results; (3) is capable of finding the global optimal solutions in most cases. However, (1) strict convergence cannot be assured; (2) they are relatively weak in terms of local search abilities; (3) in multi-modal problems, they are prone to obtain local optima [23].

2.4. Research Activity

There are two schools of thought for improving the efficiency of population-based optimization. One focuses on balancing the explore and exploit strategy with many variations, such as the elitist strategy found in some GAs. The other seeks simplification, as decisions cannot be well understood, especially for large search spaces, discontinuities, noise, or algorithms with time-varying parameters, such as PSO. The revision of the research activity is guided by the following goals:

- to find which metaheuristic among some GAs and SI algorithms performs better and discover possible ways for the system to automate the decision among them;

- to study multi-objective optimization in the transient time at the startup of HVAC systems;
- to include new optimization objectives for enhancing the lifecycle of the equipment and specifically to facilitate the transition to the steady state.

Sharif et al. [31] included the assessment of the lifecycle cost (LCC) in addition to the energy consumption and environmental impact as a new optimization objective in the passive and active building design with a GA. They managed conflicting objectives such as renovating the envelope (passive structure) or the systems (active structure). Lee [32] also combined a GA with Computational Fluid Dynamics (CFD) for the building geometry (passive design) and the HVAC system (active design), having the temperature, energy consumption, and the Index of Air Quality (IAQ) as objectives.

Gagnon et al. [33] compared the computational resources spent in sequential and holistic approaches of a net-zero building design, using NSGA-II to optimize the carbon footprint, lifecycle cost, and thermal comfort. The experiment proved that the holistic approach achieved 59% of the optimal solutions in 100 h, and the sequential approach achieved 41% in 765 h.

The work of Haniff et al. [34] is representative of introducing minor changes to an algorithm that improves the addressed problem. They modified the Global PSO so that it can outperform the optimization of the energy consumption and the temperature, considering the weather forecast, an estimation of the characteristics of the building, and the Predicted Mean Vote (PMV) for Air Conditioning scheduling.

Cai et al. [35] proposed hybridizing a multi-objective evolutionary algorithm with a quantum-behaved PSO after dividing the problem into subproblems with Tchebycheff's decomposition: Decomposition-based Multi-Objective Binary Quantum-behaved Particle Swarm Optimization (MOMBQPSO/D). The algorithm minimizes the temperature mean and deviation in area-to-point heat conduction.

Zhai et al. [36] enhanced MOGA for the secondary cooling process in continuous casting by dynamically tuning the mutation and crossover operators with the probability method. They compared it with MOPSO and MOGA and showed a 10% water reduction.

Oliva et al. [21] reviewed different metaheuristics-based algorithms applied to the estimation of solar cell parameters. They outlined the advantages and disadvantages of the GA, Harmony Search (HS), Artificial Bee Colony (ABC), SA, Cat Swarm Optimization, Differential Evolutionary, PSO, Advanced Bee Swarm Optimization, Whale Optimization Algorithm (WOA), Gravitational Search Algorithm, Flower Pollination Algorithm, Shuffled Complex Evolution, and Wind-Driven Optimization. They concluded that WOA performs better than the others regarding the accuracy and convergence speed and avoided local minima trapping.

Aguilar et al. [37] proposed a new flexible architecture for Building Management Systems (BMSs), with an Autonomic Cycle of Data Analysis Tasks (ACODAT) that makes use of banks of optimization algorithms for HVAC system control and hinted at its use for supervisory and self-optimization tasks. In fact, in a later study, they developed a Fault Detect and Diagnosis (FDD) system optimized with MOPSO, also capable of long-term equipment degradation, using the COP [15].

Awan et al. [17] analyzed the design of a solar tower plant using fuzzy goals with PSO, showing significant improvements in most of the design parameters (solar multiple, tower height, and others).

Afzal et al. [38] compared the results of applying Fuzzy Logic (FL) in both a GA and PSO to optimize the Nusselt number, friction coefficient, and maximum temperature of a battery thermal management, observing that GAs provide better results, though they are less widespread than PSO.

Suthar et al. [39] compared NSGA-II, NSGA-III, and MOPSO, applying the Technique for the Order of Preference by the Similarity to Ideal Solution (TOPSIS) for tuning

the parameters of a 2 Degree-of-Freedom (DoF) controller: the setpoint track, flow variation, and input fluid. The performance was measured with IAE, ISE, ITAE errors, and the execution time, and the step function reaction was analyzed.

Waseem Ahmad et al. [9] assessed several optimization methods and indicated that GAs perform global searches well but show poor convergence. Swarm-based algorithms are good for local searches but are slower than genetic algorithms for global searches. However, Ant Colony Optimization (ACO) is faster at searching compared to others and at converging compared to simple genetic algorithms. In an HVAC system's control, the most studied multi-objective optimization techniques are GAs, in 29% of the related literature, and MOPSO, in 10%. MOGA also stands out among them.

Behrooz et al. [40] confirmed that GAs provide optimization for comfort and energy savings because of their good behavior with nonlinear systems but are challenged with variable context information and perturbances [41]. They are sometimes combined with fuzzy control [8].

Previous and current research does not fully cover the topics addressed in this article, which constitutes a novelty. Most of the studies demonstrate GA and SI optimization in HVAC systems in both design and operations, but few compare them. Some research deals with dynamic adaptation, such as dynamic PID tuning, but none of them include optimization of the COP to enlarge the lifecycle and the rate of change in ambient temperature at the end of the transient state to moderate the damping to a steady state. Table 1 shows all cited works related to this section.

Table 1. This research's topics addressed in the cited articles.

Topics Addressed	References
HVAC system applied research	[9,17,32,34–37,39–40]
Improving operations	[9,15,34–40]
New optimization objectives	[15,31–33,35,37,39]
Genetic and swarm comparison	[9,21,36,38]
Algorithm improvements	[32,34,35,38,39]
Dynamic objectives	[35,39,41]

3. Materials and Methods

3.1. Teatro Real: The Opera House of Madrid

The case study is the HVAC system of the emblematic Opera House of Madrid (Spain), known as Teatro Real. The building has a floor size of 65,000 m² (700,000 ft²) in 10 levels above the ground and 6 underneath. The 1430 m² (15,400 ft²) stage includes the most advanced scenic technology and hosts opera and concerts for 1746 seated people in the stalls, the boxes, the balcony, and the paradise areas. The building has 11 lounges, four rehearsal rooms, and seven studios, and the scenic 'box' is surrounded by offices, warehouses, and technical premises. Figure 1 is a recent photo of the building.



Figure 1. Main façade of the Opera House of Madrid. Courtesy of Fundacion del Teatro Real.

The Opera House is open from September to July and closed in August every year. Madrid climate changes abruptly with cold winters, with an average of 0 °C (32 °F), and hot summers, with an average of 35 °C (95 °F), requiring heating and cooling. Teatro Real is also used out of the shows for rehearsals, celebrations, and product launches, making the HVAC operation a complex task.

The HVAC system of Teatro Real is an iconic example of a heterogenous HVAC system built with several refurbishments, allocating two 195 kW water–air heat pumps for both heating and cooling, and two 350 kW water–water chillers for extra cooling, managed with the same BMS. There is also a boiler and an ice accumulator that are falling into disuse.

The database provided by the Administration of Teatro Real contains historical data registered in the BMS between 1 January 2016, and 4 June 2018.

3.2. Selection of the Optimization Algorithms

The selection of the multi-objective optimization algorithms for HVAC analyzed in this study is based on the observations of Ekici et al.'s comprehensive review [42]. The initial selection of evolutionary algorithms is MOGA, NSGA-II, NSGA-III, and SPEA2.

3.2.1. The Multi-Objective Genetic Algorithm (MOGA)

Fonseca et al. [27] proposed in 1993 to compute the fitness of each individual as a weighted sum of the objective functions with random weights to obtain the probability to either select or discard it. MOGA yields interesting results, but it is not yet widely spread in real building HVAC systems.

3.2.2. The Non-Dominated Sorting Genetic Algorithm Version 2 (NSGA-II)

Deb et al. [43] proposed in 2002 to sort the individuals into categories based on non-dominance. Thus, the non-dominated individuals are in the first category. The individuals dominated by others in upper levels belong to the second and next categories. Figure 2 shows how the algorithm works.

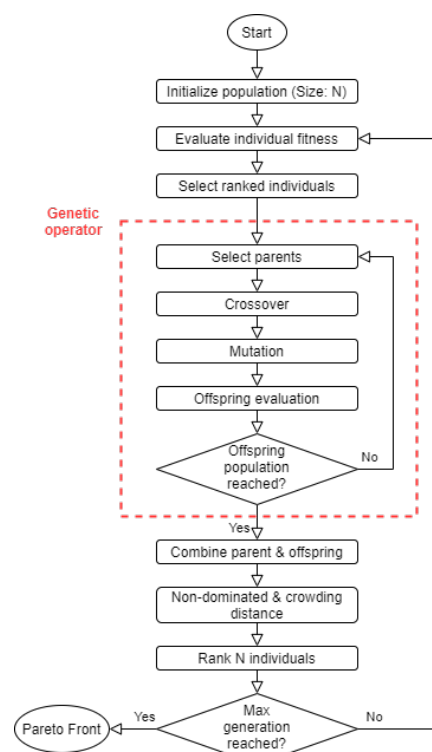


Figure 2. NSGA-II algorithm flowchart.

At the end of each iteration, the algorithm computes the distances among the individuals, known as crowding distance, for ranking.

3.2.3. The Non-Dominated Sorting Genetic Algorithm Version 3 (NSGA-III)

NSGA-III is a variant of NSGA-II that Deb et al. proposed later in 2014 [44] with an adaptive selection of the operator and a set of pre-specified (or manually) points of reference that generate a hyper-plane that improves the diversity of the population. It is conceived for improving performance when the number of objectives is larger.

3.2.4. The Strength Pareto Evolutionary Algorithm Version 2 (SPEA2)

Zitzler et al. [45] proposed in 2001 a fitness function to sort the individuals by identifying how many were dominated by a given solution and how many dominate it. The density is estimated with the k-Nearest Neighbor (k-NN) technique that prunes the elitist set (non-dominated) so that the algorithm delivers the desired number of solutions. Figure 3 shows how SPEA2 works.

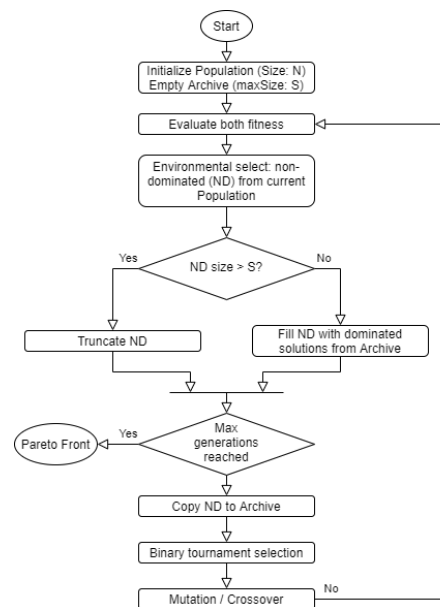


Figure 3. SPEA2 algorithm flowchart.

The other side of this analysis considers the SI-based algorithms, OMOPSO and SMPSO.

3.2.5. Optimized Multi-Objective Particle Swarm Optimization (OMOPSO)

OMOPSO is one of the MOPSO versions proposed by Reyes-Sierra et al. [46] in 2006 that uses Pareto's non-dominance to identify the leaders and the crowding distance to regulate the maximum number of them. Each iteration proclaims a leader, modifying the speed of the rest to head for it. The leaders of the current generation are set apart from the global leaders. The algorithm splits the population into groups with different mutation operators. Figure 4 shows how these algorithms work.

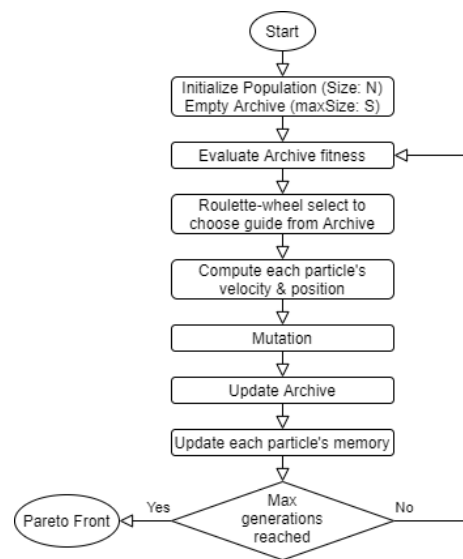


Figure 4. OMOPSO algorithm flowchart.

3.2.6. Speed-Constrained Multi-Objective Particle Swarm Optimization (SMPSO)

SMPSO, proposed by Nebro et al. in 2009, is another version of MOPSO [47] that includes a speed constraint mechanism for each individual, being good when individuals are excessively accelerated. The optimization is no-preference, bringing an important Degree of Freedom (DoF) for making tactic and strategic decisions. The result consists of “nondominated” solutions located in the hyper-plane of optimum values or the Pareto Front (PF). Thus, for instance, the operation can take optimal values increasing the ventilation to reduce the risk of transmission of disease, e.g., COVID-19, or aiming toward maximum comfort, allowing the manager or the system to pick up the best value of the PF to accomplish the goal.

In any case, diversity is preserved by either the density estimation or truncation. Fitness with the k-NN of the *i*th individual, $F(i)$, is computed as

$$F(i) = R(i) + D(i)$$

When $F(i) < 1$, the individual is non-dominated. $R(i)$ is the raw fitness, obtained from

$$R(i) = \sum_{j \in (\text{Population} + \text{Archive}), j > i} S(j)$$

where $S(j)$ is the strength value, representing the number of solutions in both Population and Archive, when i dominates:

$$S(i) = \{j / j \in (\text{Population} + \text{Archive}) \wedge i > j\}$$

$D(i)$ is the density that allows the discrimination between individuals with identical fitness values, and it is obtained from

$$D(i) = \frac{1}{\sigma_i^k + 2}$$

$$k = \sqrt{|\text{Population}| + |\text{Archive}|}$$

where σ_i^k is the distance in the objective space to the *k*th nearest neighbor in both Population and Archive. In the case of truncation,

$$i \text{ is removed, if } i < j, \forall j$$

The performances of these metaheuristics are compared with Random Search acting as a baseline, for not having any specific speeding up mechanism for exploring and exploiting the decision space.

3.3. Selection of Metrics

The “no free lunch” theorem is applicable for assessing the optimization [9], as the improvements on one feature reduce the effectiveness on another. The algorithm performance is a balance between the achievement of solutions with values close to the PF and the runtime resources required. This proves the algorithms empirically. Riquelme et al. [48] identified up to 54 metrics to prove (1) the cardinality or the number of solutions in the approximation set; (2) the accuracy, convergence, or distance to the PF; and (3) the diversity, which measures the distribution of the fitness values and how they spread. Another classification of metrics is given by the generic definition of Zitzler et al. [49], being unary if only one approximation set is received and binary if two are received. This analysis takes the top three metrics in the ranking and the one that records the runtime [48]:

- Hypervolume (HV), S metric, or Lebesgue measure: a unary metric that obtains the total space covered by the found solutions or approximation set using a reference point [11]. It considers accuracy, cardinality, and diversity.
- Generational Distance (GD): the average Euclidean distance between the approximation set with the nearest member of the ideal PF [50]. It only considers the accuracy.
- ϵ -Indicator (EI): a binary indicator that gives a factor by which an approximation set is worse than another considering all objectives.
- Execution Time (ET) or runtime: the time consumed by the optimization algorithm to fully complete the task.

3.4. Auxiliary Tools

The simulation was coded in Python, using basic NumPy, Pandas, and Datetime libraries for managing vectors, matrices, and time series. The simulation module, RFR, is implemented with Scikit recommended for machine learning [51]. The optimization is built with the JMetalPy framework [52], well proved for solving multi-objective optimization problems with metaheuristics [41]. The visualization of the obtained results is built with Matplotlib.

4. Problem Formulation

The HVAC system of Teatro Real is set to follow the mechanical and comfort set-points required for a near event. The time spent to climatize and several HVAC parameters are those that the chiller’s manufacturer initially recommended just after installation. The BMS sends commands to the HVAC system to start/stop the chillers in a certain sequence to ensure that, at the time of the event, the comfort parameters will be appropriate.

The proposed control loop for the multi-HVAC system performance optimization is depicted in Figure 5.

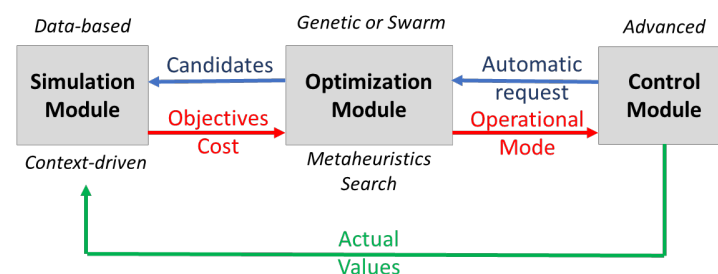


Figure 5. Advanced control optimized with a predicting context-driven model.

The Control Module, with the same functions as today, initiates the process by requesting the Optimization Module for instructions to improve its operation. The Optimization Module, which performs a metaheuristic search in the space of possible solutions, returns the best candidate obtained with the algorithm used in each model run (either

with GA or SI). The fitness functions of the candidates are evaluated by the Simulation Module that receives every individual of the population and performs the simulation of the HVAC behavior (non-linear system) [53], as defined by the candidate control parameters. The simulation is carried out with an ML algorithm, specifically a Random Forest Regressor (RFR), previously trained with historical data from the database of Teatro Real, by minimizing the Mean Squared Error (MSE) and maximizing the coefficient of determination (R^2). The RFR also requests contextual information to compute the simulation, which is provided by external sources. Finally, the Control Module translates the optimal recommendations into instructions for the actuators.

Each experiment carried out in this study executes one control cycle (petition) and addresses the optimization, without delving into the control stage. Inspired by the ACO-DAT management architecture for HVAC systems [37], an autonomous cycle updates the model offline, maintaining its accuracy in real operational conditions, as shown by the green arrow in Figure 5.

The primary objectives are to maximize comfort and minimize the consumed energy.

$$\text{Comfort} = |T_0 - T_r|$$

$$E = \sum_1^n E_i$$

where T_0 is the setpoint temperature, and T_r is the indoor room temperature, both in $^{\circ}\text{C}$. The maximum comfort for the optimization is therefore 0. The consumed electrical energy, E , is the sum of the consumed energy in kW.h in each chiller group, the multi-HVAC concept [37]. N is the number of chiller groups. The energy of one chiller group, E_i , is

$$E_i = E_{\text{chiller},i} + E_{\text{CT},i} + E_{\text{cwp},i} + E_{\text{wpp},i}$$

where $E_{\text{chiller},i}$ is the energy consumed in the chiller machine, $E_{\text{CT},i}$ is that in the cooling tower, $E_{\text{cwp},i}$ is that in the cooling water pump, and $E_{\text{wpp},i}$ is that in the chilled water primary pump.

This study includes two new objectives in the optimization as a novelty. The first one is the Coefficient of Performance, COP. The higher the COP is, the better the performance of the equipment, resulting in better energy efficiency and lower maintenance costs:

$$\text{COP} = \frac{W}{P}$$

The COP is the engineering ratio of the supplied thermal power, W , to the consumed electric power, P . The optimization of the COP brings two important advantages for the HVAC system. HVAC equipment is designed to work at maximum performance, and in this regime, the system obtains its best energy efficiency. With the appropriate autonomous cycle of data tasks [8], the supervisory system detects the degradation of the system, providing predictive maintenance [15].

The second novel objective is the rate of change in the ambient temperature, \dot{T}_r , that is the rate at which the temperature varies when it reaches the setpoint. This objective leads the system to rapidly reach the steady state with convenient damp that minimizes the overshooting:

$$\dot{T}_r = \frac{dT_r}{dt}$$

This parameter is important at sudden startups when there is a transient time before the steady state [16]. The lower the slope of the derivative is, the less impact on overshooting and steady noise there is on the next control phase.

The optimization requires Comfort, E , and \dot{T}_r to be minimized and COP to be maximized. The decision space is formed with the chillers' capacities, C_i [%], the setpoint, T_0 [$^{\circ}\text{C}$], and the schedule or the date and time at which the system is expected to reach the setpoint, t_{start} . The indoor ambient temperature when the system starts, $T_r(t=0)$ [$^{\circ}\text{C}$], the

number of occupants, N , and the outdoor ambient temperature, OAT [$^{\circ}C$], are the contextual information that determines the system. This study uses the capacity of the chillers as actuators on the subsystems, and this is justified with this simplified model:

$$P_i = P_{max} C_i$$

where P_i is the electrical power actually supplied from the i th chiller, and P_{max} is the maximum power of the chiller. The chiller's thermal power is generated according to the machine performance that is added to the other chillers, W_{HVAC} .

$$W_i = COP_i P_i$$

$$W_{HVAC} = \sum_1^n W_i$$

Thermal power conditions indoors compensate for the outdoor weather conditions and the corporal temperature of the occupants:

$$W = W_{HVAC} + W_{SUN} + W_{OCC}$$

The thermal energy, Q , is then obtained from the power, and T_r is obtained from ΔT_r , the indoor temperature variance.

$$Q = \int_0^{t_{end}} W dt$$

$$Q = C_e m \Delta T_r$$

Figure 6 shows the model with the inputs required, grouped in controllable and control variables, and the outputs, differentiating the normal optimization objectives of the thermal inertia for the next control plan [37].

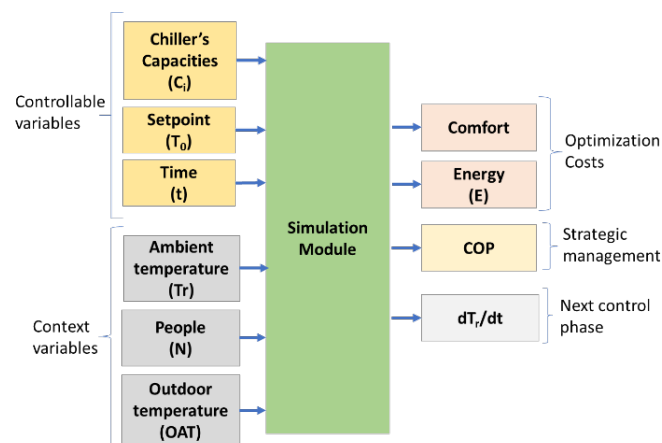


Figure 6. Simulation module's functionality to compute the cost functions for the optimization.

An individual in the population consists of a sequence of four operational modes of the chillers based on their capacities, C_i [%], at certain times, t_i , before the event starts at t_{end} [37]. Each operational mode is a 5-tuple consisting of the proposed capacities for the four chillers ranging from 0% to 100% and the time that they start. Thus, a single individual contains four of these 5-tuples. The RFR performs a simulation for each 5-tuple, chaining them according to their start-up time. The last 5-tuple indicates the operational values applied to the chillers until the system reaches the steady state, t_{end} .

The multi-objective optimization problem would be formally defined as follows:

1. Find the vector \bar{x} in the decision space:

$$\bar{x} = \begin{bmatrix} t_1 & t_2 & t_3 & t_4 \\ C_1^1 & C_1^2 & C_1^3 & C_1^4 \\ C_2^1 & C_2^2 & C_2^3 & C_2^4 \\ C_3^1 & C_3^2 & C_3^3 & C_3^4 \\ C_4^1 & C_4^2 & C_4^3 & C_4^4 \end{bmatrix}$$

t_i , where $i = 1, 2, 3$, and 4 , represents the starting dates and times to configure the capacities of every subsystem, while C_j^i , where $j = 1, 2, 3$, and 4 , represents the capacities of the chiller j during the period that starts at i and ends at $i + 1$. The last period is between t_4 and t_{end} .

2. \bar{x} will satisfy these inequality constraints at the following point:

$$|C_j^i| \leq 100$$

$$t_{i+1} \geq t_i$$

3. \bar{x} will optimize the vector function $\bar{f}(\bar{x})$ in the objective space:

$$\bar{f} = \begin{bmatrix} \text{Comfort}(\bar{x}) \\ E(\bar{x}) \\ \text{COP}(\bar{x}) \\ \frac{dT_r}{dt}(\bar{x}) \end{bmatrix}$$

Comfort and COP must be maximized, while consumed energy, E , and the rate of change of ambient temperature, T_r must be minimized.

5. Results

5.1. Dataset

The BMS is connected to 1824 digital and analog sensors, prompting the ambient and return temperatures, frozen water flow rates, valve states, chiller's performance, secondary circuit values, air flow rate, fan speeds, pumps rotational speeds, controller status, etc., and allows the operator to send instructions to the actuators from the centralized platform. However, the historical data only keeps 169 variables: outdoor temperature, room temperatures, electrical supplied power, thermal energy generated by each of the four HVAC subsystems, and their COPs grouped in several tables with different sampling rates (10 min, 15 min, 1 h, daily). Usable records are from January 2016 to June 2018. The data have been cleaned to improve the accuracy by removing nonessential fields, records with outliers, nulls, and/or zeros, getting 9898 (80%) registers for training and 2475 (20%) for validation.

The Department of Engineering prepares the work order, based on the HVAC operational mode (HOM) for the field operators based on the events schedule and the weather forecasts, and consisting of pre-programmed routines. This is, however, inefficient because the complexity of the system operation reduces all possible variations to a small set of HOMs, based on the primitive recommendations of the installers. The occupancy of the building can reach up to 1700 during performances, while the number of people on labor days is around 600.

5.2. Data Model

The multi-objective estimation is computed with the RFR with good accuracy and speed balance. The model simulates the outputs in intervals of 15 min, which is a tradeoff between the system inertia and the discretization of the system dynamics. The model receives the time required for starting up the HVAC system, t_0 , the time of the venue or the moment in which the room temperature, t_{end} , must reach the setpoint, T_0 , the room temperature at the beginning, T_r ($t = t_0$), the number of people, N , and the outdoor temperature forecast, which is a vector of temperatures from t_0 to t_{end} every 15 min. Table 2 represents an example where the temperature at 17 °C must reach the setpoint, 23.5 °C, in an hour.

Table 2. Control request & context data.

Feature	Value
HVAC startup time, t_0	18:30
Event start time, t_{end}	19:30
Setpoint, T_0	23.5 °C
Ambient Temperature, $T_r(t_0)$	17 °C
Outdoors Temperature vector, \overline{OAT}	[14 14.5 14.7 14.6]

The model also requires the outdoor temperature from the weather forecast. The optimization algorithm then releases the proposed individual for fitness.

In addition, the simulation receives the set of HOMs searched by the algorithms that will work in each interval. The algorithm is a sequence of HOMs proposed for the slots in the interval from t_0 to t_{end} , consisting of the power capacities of each chiller. Following the example, Table 3 shows one of these candidate solutions.

Table 3. Individual consisting of a sequence of four operational modes.

Time	C_1	C_2	C_3	C_4
18:30	-30	-21	0	-3
18:45	-27	-20	-10	-4
19:00	-32	-15	0	-10
19:15	-28	-20	0	-10

A negative capacity indicates that the chiller is cooling, while a positive one indicates that it is heating. Real implementations will impose restrictions that are not considered here, such as smoothing the capacity transitions from one slot to another or preparing the chiller for cooling or heating modes. Table 4 depicts the result of the optimization for this example.

Table 4. Model prediction applying optimal operational modes.

Feature	Value
Comfort	0.12 °C
Consumed Energy, E	1300 kW.h
Coef. of Performance, COP	4.5 kW/kW
Ambient Temp. Change Rate, $\frac{dT_r}{dt}$	0.15 °C/min
Time before performance	15 min

5.3. Algorithm Analysis

The analysis compares the performance and execution time (ET) of the algorithms. They start with the same expected number of solutions, i.e., the population size for the GA and the swarm size for the SI algorithm. The experiment involved population/swarm sizes ranging from 100 to 350 in steps of 50. The mutation probabilities were the same, and the SBX crossover probabilities and distribution index were the same for the GAs. The mutation scheme followed a polynomial probability distribution, except for OMOPSO, which combined uniform and nonuniform distributions with the same perturbation index, 0.5.

The algorithms stopped after 5000 iterations, and GAs stopped earlier if they were triggered with the dominance threshold. In order to obtain stable results, the algorithms were proved 10 times to determine the average of the obtained values. Figures 7–9 represent the objective space for the variables Comfort, Consumed Energy, and COP in 2D diagrams.

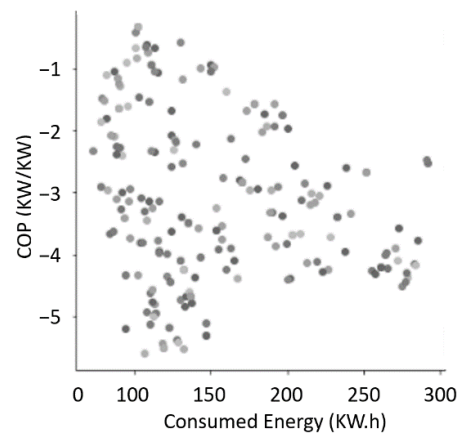


Figure 7. 2D objective space of COP [kW/kW] vs. the chiller’s Consumed Energy [kW.h].

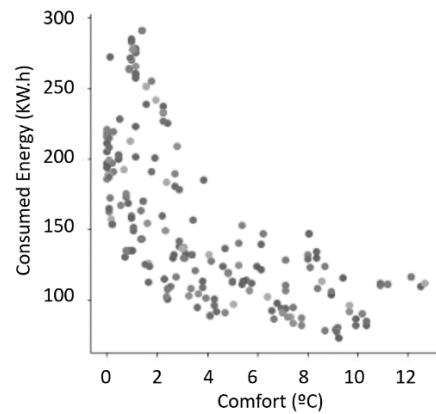


Figure 8. 2D objective space of Consumed Energy [kW.h] vs. Comfort [°C].

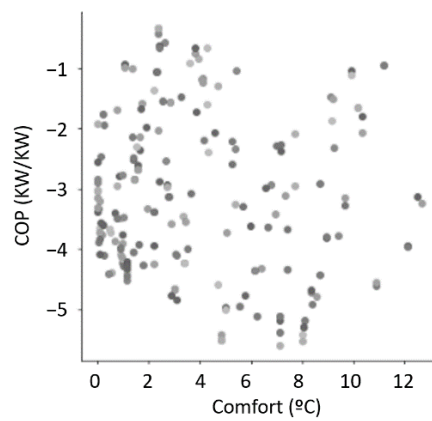


Figure 9. 2D objective space of COP [kW/kW] vs. Comfort [°C].

Metrics used in the comparison were computed with the JMetal framework, and the ETs were recorded. The SPEA2 algorithm was dropped from the analysis, as it takes 22-fold more runtime than MOGA [45]. Figure 10 shows the obtained ET values.



Figure 10. Average execution time for each algorithm.

The GAs ran faster than the SI-based algorithms. MOGA improved the Random Search by 13%, and OMOPSO improved it by 9%. It was observed that NSGA-III takes more time than NSGA-II to execute. This is because of the extra computation required for the adaptive operator and the generation of hyperplanes. On the other hand, the speed constraint mechanism seemed to increase the ET of the SMOPSO, compared with OMOPSO. All outperformed RS.

GD showed how close the fitness of the set of solutions was from the ideal PF, and this is depicted in Figure 11.

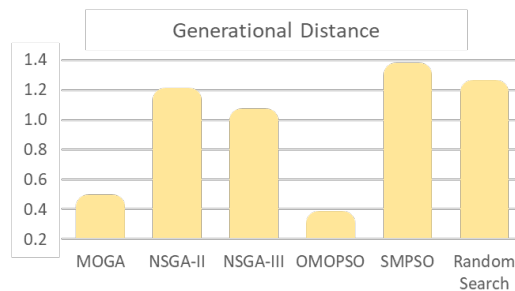


Figure 11. Average GD to the Pareto's ideal front by each algorithm.

An approximate PF was constructed running the NSGA-II 20,000 times, simulating a limit behavior. The accuracy of OMOPSO and MOGA with 75% and 65% improvements compared to Random Search was observed. It was unavoidable that the quasi-ideal PF construction was insufficient for the rest of the algorithms. HV and EI are shown in Figures 12 and 13.

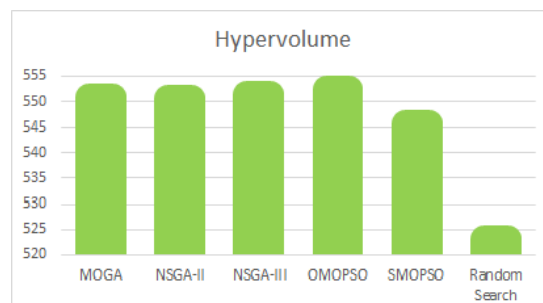


Figure 12. Average hypervolume of each algorithm.

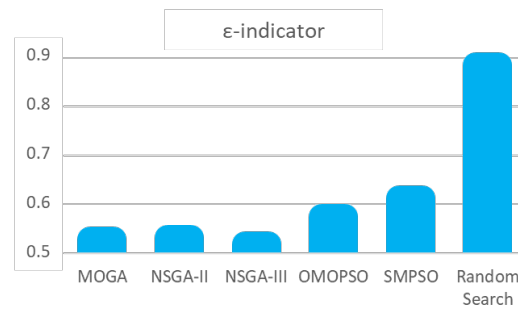


Figure 13. Average ϵ -Indicator of each algorithm.

In both metrics, it is possible to identify the significant improvements of all the algorithms compared with Random Search. The ϵ -Indicator show NSGA-III and MOGA as the best algorithms, outperforming Random Search by 42% and 40%, while the SI algorithms were worse (31–35%). The HV does not show significant differences among algorithms but shows an improvement of 5% on average.

5.4. Visualization

Regarding the question of whether an algorithm outperforms another with a combination of any quality measures, such as those seen above, Zitzler came to the conclusion that there was no such combination, but it could be seen as the equivalence to the concept of dominating [54]. Thus, Figures 14–16 show 2D maps formed with the metrics of this study, those closer to the bottom left corner being the most appropriate. The best algorithms are found in the lower-left corner in all cases. The charts also show the distance among them, presenting an intuitive method with which to make decisions as to which performs better. Figure 14 shows the behavior of the algorithms when setting the priority in ET and GD.

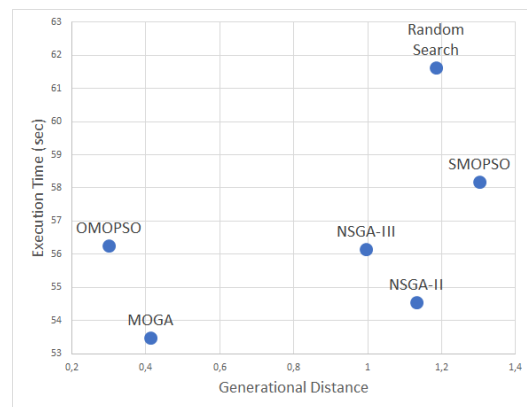


Figure 14. Plot chart mapping the studied algorithms according to the ET and the GD.

This case yields the selection of either MOGA or OMOPSO algorithms as the best for optimization accuracy. Both metrics penalize SMPSO, which obtains a GD even worse than RS. Figure 15 prioritizes the HV (the inverse in this case for obtaining a homogeneous visualization) with the ET.

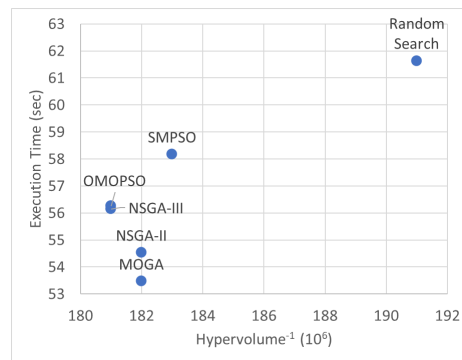


Figure 15. Plot chart mapping the studied algorithms according to the ET and the HV⁻¹.

In this case, SMPSO still performs worse than the others in terms of accuracy, but much better than RS, likely due to diversity. All the rest behave similarly, the GA family standing out. Figure 16 prioritizes the ϵ -Indicator and ET.

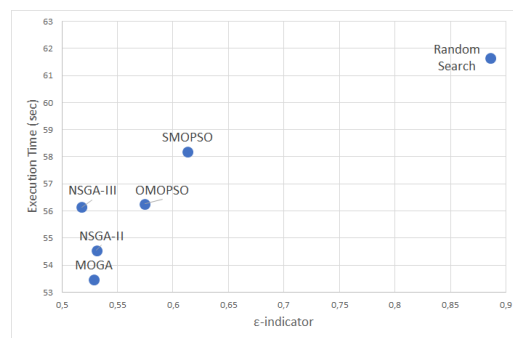


Figure 16. Plot chart mapping the studied algorithms according to the ET and the ϵ -Indicator.

ϵ -Indicator also measures the cardinality and maintains SMPSO at the back, followed by OMOPSO, while GAs shows better behavior.

5.5. Energy Efficiency Improvements

To complete the experiment, four differentiated events available in the historical data of the building were randomly selected to compare the performance of the HVAC equipment in terms of energy efficiency, with the results that would have been obtained by applying the proposed optimization. This indicates what can be expected from this approach. The events are defined in Table 5.

Table 5. Model prediction applying optimal HOMs.

Dataset cases	Event 1	Event 2	Event 3	Event 4
Date	27 September 2017	30 December 2017	29 July 2017	16 February 2018
Outdoors Temperature	35.87 °C	15.12 °C	35.51 °C	14.99 °C
Simulation Start Time	18:00	16:30	18:00	16:30
Performance Schedule	21:00	19:30	21:00	19:30
Ambient Temp. (start)	25.85 °C	23.03 °C	26.03 °C	22.82 °C
Ambient Temp. (end)	25.54 °C	23.41 °C	25.66 °C	23.26 °C

To illustrate the example, a second decision-making process with a weighted sum was set to select one of the solutions with values in the PF. Weights slightly favored Consumed Energy savings over the others. Table 6 shows the results.

Table 6. Results obtained with MOGA optimization and comparison with real data.

Objective Space	Historical Data	MOGA	Improvement
Event 1			
Consumed Energy	1138.69 KW.h	154.63 KW.h	86%
Comfort	2.04 °C	0.13 °C	<0.5 °C
COP	3.29	3.25	>3.00
Event 2			
Consumed Energy	365.25 KW.h	132.51 KW.h	64%
Comfort	−0.08 °C	0.27 °C	<0.5 °C
COP	4.31	3.42	>3.00
Event 3			
Consumed Energy	931.02 KW.h	126.85 KW.h	86%
Comfort	2.16 °C	0.20 °C	<0.5 °C
COP	3.59	4.69	>3.00
Event 4			
Consumed Energy	338.04 KW.h	132.44 KW.h	61%
Comfort	−0.23 °C	0.40 °C	<0.5 °C
COP	3.64	3.69	>3.00

The right column shows the theoretical energy savings in each case with the optimized HOMs compared with what was actually recorded in the dataset. This column also stresses the achievements in comfort with expected deviations of less than 0.5 °C and HVAC subsystems working with COPs above 3.00, which is considered a good value. These impressive results of 60–80% in energy savings, preserving the comfort and the system performance, must be adjusted with further research considering real restrictions, but they hint toward a promising line of research.

5.6. Comparison with Other Works

Several authors have proposed comparisons between NSGA-II and MOPSO, which may contribute to the comprehension of the results. Keshavarz et al. [55] compared NSGA-II and MOPSO for the stochastic optimization of an inventory control system, showing that NSGA-II has better performance in spacing and in the number of Pareto optimal solutions, while MOPSO better spreads the fitness of the solution set and consumes fewer computational resources. Niyomubyeyi et al. [56] studied optimization in evacuation planning, obtaining better convergence and spread with MOPSO, but the algorithm execution took five times longer than NSGA-II. Saldanha et al. [18] obtained similar results in convergence and spread for MOPSO and NSGA-II, although MOPSO yielded better results in spacing. Elgammal et al. [57] studied the integration of hybrid wind photovoltaic and fuel cells, obtaining similar system operating costs with both, but in this case, the MOPSO execution time was shorter than NSGA-II.

6. Conclusions

This study shows the performance of several genetic and SI-based algorithms when optimizing the control of a building HVAC system. The study works with the real historical data of a complex and singular building by adapting the control logic to the available sensed measures and individual chiller actuators. The results yield that simple MOGA and NSGA-II/III run faster than MOPSOs, confirming the pure Random Search algorithm as the slowest. The best convergence is obtained with OMOPSO according to GD and HV.

The achievement on energy consumption is impressive, as shown with several events randomly selected from the data, reaching savings from 60% to 80%. These results will be proved for generalization purposes with further research that will include the new model's restrictions.

This study is the first to take two new objectives into the optimization problem: the HVAC subsystem's performance, COP, and the rate of change in ambient temperature at the end of the system startup stage. The first objective brings the possibility of advanced supervisory policies that improve the maintenance of the equipment and extend its lifecycle. The minimization of the second allows for a smooth transition to the permanent stage of the HVAC operation, reducing the overshoots or the underdamping effect of the room temperature values. In the following works, the dominance variation produced when adding new conflicting objectives and how this affects control system decision-making will be analyzed.

The proposed simple visualization of the algorithms not only allows for an intuitive understanding of which algorithm performs better but also opens the possibility of the automatic real-time instantiation of the most convenient algorithm from a bank of optimizers according to given contextual information. This is important because there are no rigid rules, but rather, existing or new strategies, such as running out of time, operations when the building is closed, etc.

The article also claims for consensus in optimization with a body of knowledge that integrates the contribution of the different disciplines that theorize or are applicable to the case.

This study requires generalization to demonstrate its scope with other different buildings, HVAC systems, and overall different variables extracted from the control logic. It is also of interest to work on parameter tuning to characterize the inherent “no free lunch” theorem.

The use of real data has made the study more reliable. The singularity of the building and the heterogeneous equipment that forms the HVAC system represents a demanding test for this research.

This research will contribute to the development of the smart city with autonomic management systems capable of learning from experience and improving with the context using AI to overcome the complexity of the managed systems and changing the user's requirements.

Author Contributions: Conceptualization, N.G.-S., J.-M.G.-P. and A.G.-J.; methodology, A.-J.G.-T.; software, A.G.-J.; validation, A.G.-J. and N.G.-S.; formal analysis, A.G.-J.; investigation, A.G.-J. and A.-J.G.-T.; resources, N.G.-S., A.-J.G.-T. and J.-M.G.-P.; data curation, A.G.-J.; writing—original draft preparation, A.G.-J.; writing—review and editing, A.G.-J., A.-J.G.-T. and J.-M.G.-P.; visualization, A.G.-J.; supervision, A.G.-J., A.-J.G.-T. and J.-M.G.-P.; project administration, A.-J.G.-T. and A.G.-J.; funding acquisition, A.-J.G.-T.. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially supported by the Vice Rectorate for Research of the Universidad Francisco de Vitoria with grant code MOGA-TR, Reference: UFV2020-34.

Acknowledgments: This work was possible thanks to the contribution of the Management Committee of Teatro Real de Madrid by providing the database of the HVAC BMS System. The authors wish to thank Raul Jiménez-Juarez for his help in preparing the code as part of his end-of-degree project in the Universidad Francisco de Vitoria.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study, in the collection, analyses, or interpretation of data, in the writing of the manuscript, or in the decision to publish the results.

References

1. Plecher, H. Global Gross Domestic Product (GDP) at Current Prices from 2014 to 2024 (in Billion U.S. Dollars). Statista. 2019. Available online: <https://www.statista.com/statistics/268750/global-gross-domestic-product-gdp/> accessed on 12 May 2020).
2. Ritchie, H.; Roser, M. Energy. Our World in Data. 2018. Available online: <https://ourworldindata.org/energy> (accessed on 12 May 2020).
3. Mauro, G.M.; Hamdy, M.; Vanoli, G.P.; Bianco, N.; Hensen, J.L. A new methodology for investigating the cost-optimality of energy retrofitting a building category. *Energy Build.* **2015**, *107*, 456–478.

4. Minoli, D.; Sohraby, K.; Occhiogrosso, B. IoT considerations, requirements, and architectures for smart buildings—Energy optimization and next-generation building management systems. *IEEE Internet Things J.* **2017**, *4*, 269–283.
5. Reynolds, J.; Rezgui, Y.; Kwan, A.; Piriou, S. A zone-level, building energy optimisation combining an artificial neural network, a genetic algorithm, and model predictive control. *Energy* **2018**, *151*, 729–739.
6. Wang, F.; Yoshida, H.; Ono, E. Methodology for optimizing the operation of heating/cooling plants with multi-heat-source equipments. *Energy Build.* **2009**, *41*, 416–425.
7. Aste, N.; Manfren, M.; Marenzi, G. Building Automation and Control Systems and performance optimization: A framework for analysis. *Renew. Sustain. Energy Rev.* **2017**, *75*, 313–330.
8. Escobar, L.M.; Aguilar, J.; Garcés-Jiménez, A.; De Mesa, J.A.G.; Gomez-Pulido, J.M. Advanced Fuzzy-Logic-Based Context-Driven Control for HVAC Management Systems in Buildings. *IEEE Access* **2020**, *8*, 16111–16126.
9. Ahmad, M.W.; Mourshed, M.; Yuce, B.; Rezgui, Y. Computational Intelligence Techniques for HVAC systems: A review. In *Building Simulation*; Tsinghua University Press: Beijing, China, 2016; Volume 9, pp. 359–398.
10. Mallawaarachchi, V. Introduction to genetic algorithms-including example code. *Towards Data Science*. 2017. Available online: <https://towardsdatascience.com/introduction-to-genetic-algorithms-including-example-code-e396e98d8bf3> (accessed on 12 May 2020).
11. Shaikh, P.H.; Nor, N.B.M.; Nallagownden, P.; Elamvazuthi, I.; Ibrahim, T. A review on optimized control systems for building energy and comfort management of smart sustainable buildings. *Renew. Sustain. Energy Rev.* **2014**, *34*, 409–429.
12. Zadeh, L.A. (1993). Fuzzy Logic, Neural Networks and Soft Computing. In *Safety Evaluation Based on Identification Approaches Related to Time-Variant and Nonlinear Structures*; Springer: Berlin/Heidelberg, Germany, 1993; pp. 320–321.
13. Yang, X.S. Nature-inspired optimization algorithms: Challenges and open problems. *J. Comput. Sci.* **2020**, *46*, 101104.
14. Afroz, Z.; Shafiullah, G.M.; Urmee, T.; Higgins, G. Modeling techniques used in building HVAC control systems: A review. *Renew. Sustain. Energy Rev.* **2018**, *83*, 64–84.
15. Aguilar, J.; Ardila, D.; Avendaño, A.; Macias, F.; White, C.; Gomez-Pulido, J.; Gutiérrez de Mesa, J.A.; Garces-Jimenez, A. An Autonomic Cycle of Data Analysis Tasks for the Supervision of HVAC Systems of Smart Building. *Energies* **2020**, *13*, 3103.
16. Aguilar, J.; Garces-Jimenez, A.; Gómez-Pulido, J.M.; R-Moreno, M.D.; Gutiérrez de Mesa, J.A.; Gallego-Salvador, N. Autonomic Management of a Building’s multi-HVAC System Start-Up. *IEEE Access* **2021**, *9*, 70502–70515.
17. Awan, A.B.; Chandra Mouli, K.V.V.; Zubair, M. Performance enhancement of solar tower power plant: A multi-objective optimization approach. *Energy Convers. Manag.* **2020**, *225*, 113378.
18. Saldanha, W.H.; Soares, G.L.; Machado-Coelho, T.M.; dos Santos, E.D.; Ekel, P.I. Choosing the best evolutionary algorithm to optimize the multiobjective shell-and-tube heat exchanger design problem using PROMETHEE. *Appl. Therm. Eng.* **2017**, *127*, 1049–1061.
19. Kuan, Y.N.; Ong, H.S. Optimization of heating, Ventilating and Air Conditioning (HVAC) systems: A review. *Univ. Tenaga Nasional.* **2018**, *13*, 9049–9906. Available online: <http://dspace.uniten.edu.my/jspui/handle/123456789/11701> (accessed on 12 May 2020).
20. Nasruddin, S.; Satrio, P.; Mahlia, T.M.I.; Giannetti, N.; Saito, K. Optimization of HVAC system energy consumption in a building using artificial neural network and multi-objective genetic algorithm. *Sustain. Energy Technol. Assess.* **2019**, *35*, 48–57.
21. Oliva, D.; Elaziz, M.A.; Elsheikh, A.H.; Ewees, A.A. A review on meta-heuristics methods for estimating parameters of solar cells. *J. Power Sources* **2019**, *435*, 126683.
22. Wang, G.G.; Shan, S. Review of metamodeling techniques in support of engineering design optimization. *J. Mech. Des.* **2007**, *129*, 370–380.
23. Nabaei, A.; Hamian, M.; Parsaei, M.R.; Safdari, R.; Samad-Soltani, T.; Zarrabi, H.; Ghassemi, A. Topologies and performance of intelligent algorithms: A comprehensive review. *Artif. Intell. Rev.* **2018**, *49*, 79–103.
24. Gao, L.; Hwang, Y.; Cao, T. An overview of optimization technologies applied in combined cooling, heating and power systems. *Renew. Sustain. Energy Rev.* **2019**, *114*, 109344.
25. Castillo-Martínez, A.; Ramon Almagro, J.; Gutierrez-Escolar, A.; Del Corte, A.; Castillo-Sequera, J.L.; Gómez-Pulido, J.M.; Gutiérrez-Martínez, J.M. Particle swarm optimization for outdoor lighting design. *Energies* **2017**, *10*, 141.
26. Wang, H.; Deutz, A.H.; Bäck, T.; Emmerich, M. Hypervolume Indicator Gradient Ascent Multi-Objective Optimization. In *EMO*; Volume 10173 of Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2017; pp. 654–669.
27. Fonseca, C.M.; Fleming, P.J. Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization. In *ICGA*; Morgan Kaufmann: Burlington, MA, USA, 1993; pp. 416–423.
28. Zhang, Q.; Li, H. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Trans. Evol. Comput.* **2007**, *11*, 712–731.
29. Kumar, D.; Kumar, S.; Bansal, R.; Singla, P. A survey to nature inspired soft computing. IGI Global. *Int. J. Inf. Syst. Modeling Des.* **2017**, *8*, 112–133.
30. Robic, T.; Filipic, B. DEMO: Differential Evolution for Multiobjective Optimization. In *EMO*; Volume 3410 of Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2005; pp. 520–533.
31. Sharif, S.A.; Hammad, A. Simulation-Based Multi-Objective Optimization of institutional building renovation considering energy consumption, Life-Cycle Cost and Life-Cycle Assessment. *J. Build. Eng.* **2019**, *21*, 429–445.
32. Lee, J. Multi-objective optimization case study with active and passive design in building engineering. *Struct. Multidiscip. Optim.* **2019**, *59*, 507–519.

33. Gagnon, R.; Gosselin, L.; Armand Decker, S. Performance of a sequential versus holistic building design approach using multi-objective optimization. *J. Build. Eng.* **2019**, *26*, 100883.
34. Haniff, M.F.; Selamat, H.; Khamis, N.; Alimin, A.J. Optimized scheduling for an air-conditioning system based on indoor thermal comfort using the multi-objective improved global particle swarm optimization. *Energy Effic.* **2019**, *12*, 1183–1201.
35. Cai, H.; Guo, K.; Liu, H.; Xiang, W.; Liu, C. Multiobjective optimization of area-to-point heat conduction structure using binary quantum-behaved PSO and Tchebycheff decomposition method. *Can. J. Chem. Eng.* **2020**, *99*, 1211–1227.
36. Zhai, Y.-Y.; Li, Y.; Ao, Z.-G. Optimization of Continuous Casting Secondary Cooling Based on an Enhanced Multi-objective Genetic Algorithm. *Northeast. Univ. Dongbei Daxue Xuebao J. Northeast. Univ.* **2019**, *40*, 658–662.
37. Aguilar, J.; Garcès-Jiménez, A.; Gallego-Salvador, N.; Gutiérrez de Mesa, J.A.; Gomez-Pulido, J.M.; Garcia-Tejedor, À.J. Automatic management architecture for multi-HVAC systems in smart buildings. *IEEE Access* **2019**, *7*, 123402–123415.
38. Afzal, A.; Ramis, M.K. Multi-objective optimization of thermal performance in battery system using genetic and particle swarm algorithm combined with fuzzy logics. *J. Energy Storage* **2020**, *32*, 101815.
39. Suthar, H.A.; Gadit, J.J. Multiobjective optimization of 2DOF controller using Evolutionary and Swarm intelligence enhanced with TOPSIS. *Heliyon* **2019**, *5*, e01410.
40. Behrooz, F.; Mariun, N.; Marhaban, M.H.; Mohd Radzi, M.A.; Ramli, A.R. Review of control techniques for HVAC systems—Nonlinearity approaches based on Fuzzy cognitive maps. *Energies* **2018**, *11*, 495.
41. Emmerich, M.T.; Deutz, A.H. A tutorial on multiobjective optimization: Fundamentals and evolutionary methods. *Nat. Comput.* **2018**, *17*, 585–609.
42. Ekici, B.; Cubukcuoglu, C.; Turrin, M.; Sariyildiz, I.S. Performative computational architecture using swarm and evolutionary optimisation: A review. *Build. Environ.* **2019**, *147*, 356–371.
43. Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **2002**, *6*, 182–197.
44. Deb, K.; Jain, H. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints. *IEEE Trans. Evol. Comput.* **2014**, *18*, 577–601.
45. Zitzler, E.; Laumanns, M.; Thiele, L. *SPEA2: Improving the Strength Pareto Evolutionary Algorithm*; TIK—Report 103; Eidgenössische Technische Hochschule Zürich, Institut für Technische Informatik und Kommunikationsnetze: Zürich, Switzerland, 2001.
46. Reyes-Sierra, M.; Coello, C.C. Multi-objective particle swarm optimizers: A survey of the state-of-the-art. *Int. J. Comput. Intel. Res.* **2006**, *2*, 287–308.
47. Nebro, A.J.; Durillo, J.J.; Garcia-Nieto, J.; Coello, C.C.; Luna, F.; Alba, E. SMPSO: A new PSO-based metaheuristic for multi-objective optimization. In Proceedings of the 2009 IEEE Symposium on Computational Intelligence in Multi-Criteria Decision-Making (MCDM), Nashville, TN, USA, 30 March–2 April 2009; pp. 66–73.
48. Riquelme, N.; Von Lüken, C.; Baran, B. Performance metrics in multi-objective optimization. In Proceedings of the 2015 Latin American Computing Conference (CLEI), Arequipa, Peru, 19–23 October 2015; pp. 1–11.
49. Zitzler, E.; Thiele, L.; Laumanns, M.; Fonseca, C.M.; Da Fonseca, V.G. Performance assessment of multiobjective optimizers: An analysis and review. *IEEE Trans. Evol. Comput.* **2003**, *7*, 117–132.
50. Van Veldhuizen, D.A. *Multiobjective Evolutionary Algorithms: Classifications, Analyses, and New Innovations (No. AFIT/DS/ENG/99-01)*; Air Force Institute of Technology: Wright-Patterson AFB, OH, USA, 1999.
51. Géron, A. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*; O'Reilly Media: Sebastopol, CA, USA, 2019.
52. Barba-González, C.; García-Nieto, J.; Nebro, A.J.; Aldana-Montes, J.F. Multi-objective big data optimization with jmetal and spark. In Proceedings of the International Conference on Evolutionary Multi-Criterion Optimization, East Lansing, MI, USA, 10–13 March 2017; pp. 16–30.
53. Zadeh, L.A. Fuzzy Logic, Neural Networks, and Soft Computing. In *Fuzzy Sets, Fuzzy Logic, and Fuzzy Systems: Selected Papers by Lotfi a Zadeh*; World Scientific: Singapore, 1996; pp. 775–782.
54. Zitzler, E.; Laumanns, M.; Thiele, L.; Fonseca, C.M.; da Fonseca, V.G. Why quality assessment of multiobjective optimizers is difficult. In Proceedings of the 4th Annual Conference on Genetic and Evolutionary Computation, New York, NY, USA, 9–13 July 2002; pp. 666–674.
55. Keshavarz, M.; Pasandideh, S.H.R. Multi-objective optimisation of continuous review inventory system under mixture of lost sales and backorders within different constraints. Inderscience Publishers. *Int. J. Logist. Syst. Manag.* **2018**, *29*, 327–348.
56. Niyomubyeyi, O.; Sicuaio, T.E.; Díaz González, J.I.; Pilesjö, P.; Mansourian, A. A Comparative Study of Four Metaheuristic Algorithms, AMOSA, MOABC, MSPSO, and NSGA-II for Evacuation Planning. *Algorithms* **2020**, *13*, 16.
57. Elgammal, A.; El-Naggar, M. Energy management in smart grids for the integration of hybrid wind-PV-FC-battery renewable energy resources using multi-objective particle swarm optimisation (MOPSO). *J. Eng.* **2018**, *11*, 1806–1816.