sustainability

MDPI

Article

Swarm Intelligence-Based Multi-Objective Optimization Applied to Industrial Cooling Towers for Energy Efficiency

Nadia Nedjah [1,*,†], Luiza de Macedo Mourelle [2,†] and Marcelo Silveira Dantas Lizarazu [1,†]

[1] Department of Electronics Engineering and Telecommunications, State University of Rio de Janeiro, Rio de Janeiro 20550-013, Brazil
[2] Department of Systems Engineering and Computation, State University of Rio de Janeiro, Rio de Janeiro 20550-013, Brazil
* Correspondence: nadia@eng.uerj.br
† These authors contributed equally to this work.

Abstract: Cooling towers constitute a fundamental part of refrigeration systems in power plants and large commercial buildings. Their main function is to treat the heat emitted by other equipment to cool down the temperature of the environment and/or processes. In the considered refrigeration system, cooling towers are coupled with compression chillers. The serious world-wide concerns with regard to environmental wear and water scarcity are now common knowledge. One way to mitigate their impact is to reach a state of maximum energy efficiency in industrial processes. For this purpose, this work proposes the application of multi-objective optimization algorithms to find out the optimal operational setpoints of the studied refrigeration system. Here, we exploit swarm intelligence strategies to offer the best trade-offs. This consists of finding solutions that maximize the cooling tower's effectiveness and yet minimize the global power requirement of the system. Additionally, the solutions must also respect operational constraints for the safe operation of the equipment. In this investigation, we apply two algorithms, multi-objective particle swarm optimization and multi-objective TRIBES, using two different models. The achieved results are compared considering two different scenarios and two different models of the refrigeration system. This allows for the selection of the best algorithm and best equipment model for energy efficiency of the refrigeration system. For the studied configuration, we achieve an energy efficiency factor of 1.78, allowing power savings of 9.48% with tower effectiveness reduction of only 5.32%.

Keywords: energy efficiency; cooling towers; swarm intelligence; multi-objective optimization

Citation: Nedjah, N.; Mourelle, L.d.M.; Lizarazu, M.S.D. Swarm Intelligence-Based Multi-Objective Optimization Applied to Industrial Cooling Towers for Energy Efficiency. *Sustainability* **2022**, *14*, 11881. https://doi.org/10.3390/su141911881

Academic Editors: Rui Manuel de Sousa Fragoso, António Manuel de Sousa Xavier and Maria De Belém Costa Freitas

Received: 29 July 2022
Accepted: 14 September 2022
Published: 21 September 2022

1. Introduction

In general, industrial processes generate heat. Often, this heat must be dealt with so it can be safely dissipated. As the main coolant, water is generally exploited to dissipate heat in refrigeration systems. However, the water used in such systems always returns at higher temperatures. It is then discarded or cooled down so that it can be reused. Whenever there is a lack of water, the second solution is always safer and more economic. The disposal of water used in a cooling process is an environmentally unsustainable practice, since the water, in these cases, usually ends at a high temperature, and its disposal under these conditions would impact the local underwater flora and fauna. Furthermore, taking into account the current challenges in obtaining potable water, the reuse of water in cooling towers is essential in terms of economic and environmental sustainability. The growing environmental concern with the inefficient use of electricity and its growing demand, along with the misuse of existing water resources, which are increasingly scarce, are leading the technical and scientific community to adopt premises and measures to achieve the maximum energy efficiency of industrial installations. Therefore, any cooling system requires the application of sophisticated mechanisms that allow achieving the best possible energy efficiency.

Currently, there is a scarcity of energy resources. In general, the availability of electricity in the energy matrix of any economy largely depends on the regular occurrence of rain in areas where hydroelectric plants are installed. When this does not happen, thermoelectric plants assume part of the energy demand. However, this alternative causes a significant increase in the cost of electricity, causing economic losses for industry in general. In addition to the scarcity of water resources for power generation, there is also the basic scarcity of water, which is used in large volumes and in various forms in the food and beverage, petrochemical, and power industries, for example. This is the case both for the purpose of directly obtaining the final product and for cooling purposes, hence the motivation behind this research work. The scarcity of water, and consequently, of electricity, causes a considerable increase in the final cost of products, and losses for the industrial and commercial sectors. This yields an even greater impact for industries or companies that do not apply strategies to reduce water and power requirements.

Cooling towers are used in refrigeration systems to cool down the temperature of the returning water so it can be reused in the refrigeration process. Thus, cooling towers are usually used in large buildings, such as commercial centers and high-class hotels. A cooling tower is usually coupled with chillers and pumps to move the water inside the refrigeration system. In addition, cooling towers include fans that allow correct circulation of the system's used water. However, the chillers consume a large portion of the whole system's energy [1]. Besides cooling towers, the considered refrigeration system includes other equipment that contributes to the system's operation. Operational parameter adjustment of one of these pieces of equipment can cause ripple effects on the operation of the other equipment. It is clear that the effects can be either positive or negative.

In practice, it is observed that the application of undue, poorly planned, or poorly informed operational adjustments tends to lead a cooling system to a poor operating point or to poor performance, causing an increase in its overall energy consumption. In general, if one optimizes the operation of a certain piece of equipment, one will unintentionally harm the operation of another. Only a global model of the system, obtained from the study and individual analysis of the operation of each piece of equipment that is part of the cooling system, can determine whether the modification of a certain operational parameter will cause an increase or a reduction in the performance of the overall system. Moreover, water consumption can also be affected by on the adjustments made, since there is always a loss of water associated with the operation of a cooling tower, due to the drag of water caused by the tower's fans. This phenomenon is described in more detail in Section 2, and is also associated with the operational efficiency of the cooling tower. Therefore, an undue speed adjustment of the cooling tower fans, for example, can cause an increase in water drag from the tower due to a reduction in its operational efficiency. Consequently, this would lead to an increase in water consumption. The operational efficiency of a cooling tower is dependent on the efficiency of the thermal exchange between the hot water that comes from the process and the air mass induced to counterflow in the tower, through the use of fans. This performance is also influenced by several climatic and operational factors. Therefore, we conclude that the use of a decision support system for the operational optimization and energy consumption minimization of a cooling system is a complex process that demands faithful and precise modeling of the equipment that interoperates. It is in this context that the use of computational intelligence techniques is viable for the application of energy optimization, based on its low implementation cost and great ability to assist in decision making. To do so, sufficient quantitative and qualitative data are needed to implement and obtain satisfactory results. These data are usually obtained from field data collection and information provided by the equipment manufacturers that compose the system to be optimized.

This research aimed to solve a multi-objective optimization problem composed of the following conflicting objectives: (i) maximization of the efficiency of the thermal exchange performed by a cooling tower; and (ii) minimization of the global electrical power requirement of a cooling system, considering all the equipment necessary for its proper

functioning. In this work, two multi-objective optimization algorithms were applied. These are based on swarm intelligence: MOPSO and MO-TRIBES. We compared their performances to those yielded by three other multi-objective algorithms: NSGA-II, SPEA2, and Micro-GA [2,3]. It was expected that these algorithms would yield optimal or near optimal settings of the operational parameters for a refrigeration system based on cooling towers, tower fans, and chillers. The parameters consisted of the cooling tower fan speed setpoint and the water temperature setpoint to be provided by the chillers. The proposed methodology to minimize energy consumption while maximizing the effectiveness of the system can be generalized to any other application. However, before it can be applied, the model, objective functions, and restrictions of the application must be adjusted accordingly. It is noteworthy to point out that the details of the modeling proposed for the cooling tower with its fans are presented in [4], and the details of the model proposed for the chillers are described in [5].

The solution proposed in this work takes into account the operational restrictions imposed by the equipment composing the cooling system. These restrictions were defined based on the information provided by the equipment manufacturers. This work considers the use of compression-based chillers, for which we exploit two different models. The results obtained for both models are compared in terms of accuracy with regard to the real field data. The two chiller's models are used to define the objective functions of the applied optimization algorithms. The results are also compared under different scenarios in terms of optimization stopping criteria. The performance results for both models and stopping criteria are compared, allowing the election of the best algorithm for each scenario and the best algorithm for the application.

This paper is structured into six sections. First, in Section 2, we introduce the structure of a cooling tower, the configuration of a refrigeration system, and the concepts that impact its energy efficiency. Then, in Section 3, we review the main related research works. After that, in Section 4, we formulate the optimization problem by defining the corresponding objective functions and operational restrictions. Subsequently, in Section 5, we describe and motivate the developed system models. Later, in Section 6, we give a brief description of the optimization methods exploited in this work. Subsequently, in Section 7, we present and analyze the results. After that, in Section 8, we present a comparison of the effectiveness and efficiency of the algorithm in optimizing the energy efficiency of the refrigeration system. Then, in Section 9, we discuss the study's final results. Finally, in Section 10, conclusions are drawn and future work directions are pointed out.

2. Refrigeration Systems and Energy Efficiency

This section aims to present the basic functional concepts of the main equipment of the studied refrigeration system, so that it becomes possible to formalize the underlying optimization problem involving the cooling tower and the compression chillers. The concepts presented herein serve as a basis for the mathematical modeling of equipment, considering its thermodynamic principles of operation.

2.1. Cooling Tower

As the name suggests, cooling towers are equipment used for cooling water. This water usually comes from heat exchangers in process industries or from condensers in the case of electric generation plants. The reuse of this water is vital for the environmental and economic sustainability of the process. Cooling towers can be classified according to the used heat transfer method, relative flow between air and water, or draft. Figure 1 describes the structure and function of the modeled cooling tower. The cooling tower's functionality consists of reducing the temperature of the inlet hot water to a reasonable degree to guarantee the correct operation of the remaining processing equipment of the refrigeration system.

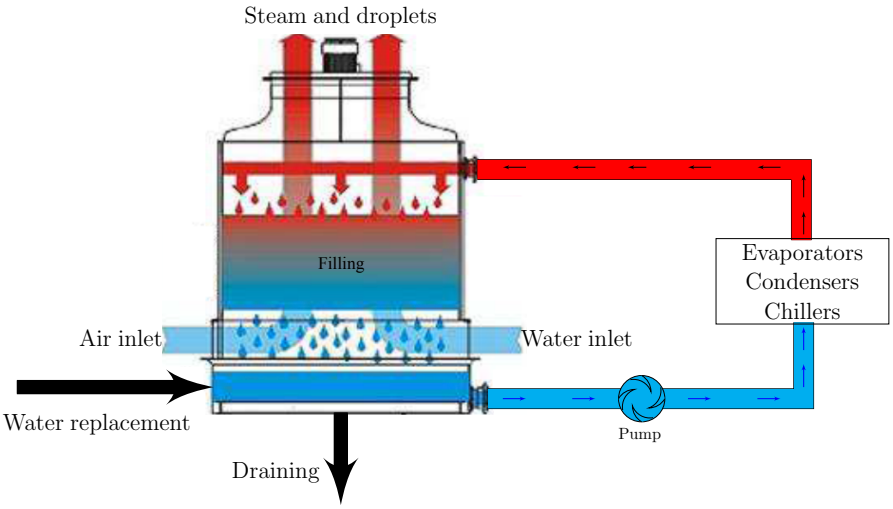

Figure 1. Cooling tower structure and function.

It can be seen in Figure 1 that there is a need for chemical treatment in the water circuits of the tower, in order to avoid corrosion and accumulation of dirt, which causes an increase in the pressure drop, causing a greater energy consumption for the water lifting pumps. Corrosion causes small leaks, which increase the tower's water consumption, leading to the need for interventions and unscheduled stops for maintenance.

A cooling tower can be classified as dry, wet, or mixed depending on the heat transfer method used. The heat exchange is considered dry when it is carried out in a closed circuit, with no direct contact between the two fluids. In this case, the fluids exchange heat through a surface, such as a metallic pipe. In wet heat exchange, there is direct contact between the fluids. This type of tower is also called an open system and uses the principle of evaporative cooling. On the other hand, the mixed or hybrid towers use the two mechanisms presented above, but there is the application of cold water (from another source) in the heat exchangers, favoring the evaporative thermal exchange. The cooling towers considered in this work are the wet type. In this case, the hot water is taken from the process to the sprinkler point, located at the top of the tower, through lifting pumps. After reaching this point, the hot water is released so that heat exchange occurs with the air flowing in the opposite direction. After performing the heat exchange, the cooled water descends through the filling to the bottom of the tower, returning to the process.

During the fall, after being released at the sprinkler point, the water comes into contact with the air stream, exchanging sensible and latent heat. It is known that part of the hot water that enters the tower evaporates. However, this water loss is not significant during operation whenever the tower's setpoint is carefully configured [6]. Additionally, a cooling tower can be classified as counter-current flow or cross-flow depending on the relative flow between water and air streams. In cross-flow towers, the air enters through the side walls and passes through the filling practically horizontally. The water falls due to gravity on both sides of the tower, crossing the air current induced by the axial fan located at the top of the tower. In counter-current towers, the air enters through the base of the tower and crosses the filling in a vertical direction, contrary to the water flow. There is a higher pressure drop in the filling compared to cross-flow towers. This causes higher fan consumption. Counter-current flow towers are more efficient, since there is less possibility of internal air re-circulation. In addition, for this type of tower, the air spends more time in contact with the water, since the air intake is carried out at the bottom of the tower, allowing more efficient thermal exchange to take place.

Moreover, the draft cooling towers can be further classified as either natural or mechanical. A natural draft tower is usually larger, i.e., about 200 m in height and 100 m in diameter. In this kind of tower, the air flows naturally due to the gradual increase in air temperature as heat dissipation occurs (convection mechanism). This is also accelerated by

their hyperbolic form. This shape causes the hot air to be accelerated along the rise, due to the area restriction along the path from the base of the tower to its top. On the other hand, mechanical draft towers are those that use fans to cause air circulation. They can be induced or forced draft. The basic difference is that in the former, the fans are positioned at the air outlet, usually at the top of the tower, whereas in the latter, the fans are positioned at ground level. The choice depends on several factors, such as available space, installation location, and average wind speed throughout the year in the region.

2.2. Compression Chiller

Chillers perform their main function of water cooling via a thermodynamic cycle. There are two types of chillers: compression and absorption chillers. The former takes advantage of a physical process, and the latter is based on physical/chemical principles. The studied refrigeration system is based on compression chillers.

Compression chillers include a mechanical compressor driven by an electric motor. It is employed to compress the refrigerant gas of the chiller. Additionally, the chiller includes other equipment. The main pieces are the condenser, evaporator, and expansion valve [7]. The thermodynamic cycle of compression chillers is described in Figure 2.

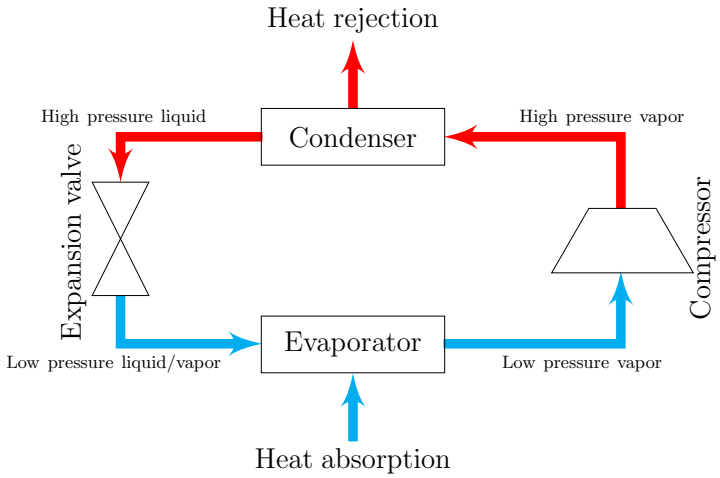

Figure 2. Illustration of the operation cycle of a compression chiller.

The evaporator allows heat exchange to occur. It is composed of tubes. On one side, there is the passage of the refrigerant fluid, and on the other, the circulation of water to be refrigerated. Thus, the evaporator cools down the water, which is lifted by the circulation pumps. The evaporation of the refrigerant fluid is a phenomenon that happens once the thermal exchange with the process outlet water occurs. Note that this water circulates in the evaporator, causing the refrigerant fluid temperature to raise. Even though this process should be isobaric, in practice, a pressure reduction in the refrigerant fluid passing via the evaporator is inevitable. The compressor is able to compensate for this reduction and sends the fluid to the condenser. Note that after its passage through the evaporator, the refrigerant fluid goes to a vapor state, which then goes through the condenser.

The condenser may be air cooled or liquid cooled. For the first type, fans are employed to permit the circulation of the refrigerant gas via finned tubes. This allows the thermal exchange between the forced air and the refrigerant gas. For the second type of condenser, the water, circulating through the employed cooling tower, allows thermal exchange with the refrigerant gas. The coils located inside the condenser facilitate this exchange, which causes the condensation of the refrigerant gas. Then, the pumps lift the hot water to the cooling tower. After this, the condensed water returns to the chiller's condenser. Note that the thermal exchange with the refrigerant vapor is a continuous process.

Moreover, an expansion/thermostatic valve is used in the chilling cycle. The valve is a device that allows direct expansion refrigeration. Thus, the fluid expansion occurs in the

environment to be chilled. Thus, the thermodynamic cycle in a compression-based chiller starts with the refrigerant fluid being compressed by the compressor in the over-heated gas state. This increases its temperature and pressure. Then, the refrigerant fluid goes through the condenser, allowing the heat to be dissipated into the outside environment. This thermal exchange causes the refrigerant gas to cool down and liquefy. The refrigerant fluid then flows via the thermostatic valve, causing a pressure reduction which in turn causes a temperature drop. Lastly, the refrigerant fluid passes through the evaporator, absorbing the water's heat and cooling it down, achieving the refrigeration goal. Then, the refrigerant fluid changes again to vapor, leaves the evaporator as super-heated vapor, and returns to the compressor, and the thermodynamic cycle restarts.

3. Related Works and Contributions

This section surveys related works. The optimization strategies used can be viewed as extreme, as reviewed in Section 3.1; evolutionary, as reviewed in Section 3.2; learning, as reviewed in Section 3.3; and swarming, as reviewed in Section 3.4. For each of these classes, we contrast the approach used in this work and present the pros and cons in each case.

3.1. Extreme Strategies

In [8], the energy efficiency of the refrigeration system was achieved through a control strategy based on extreme search. The simulated process consisted of a water-cooled screw-type chiller and a wet-type cooling tower with a counter-current mechanical draft. The dynamic models of the equipment were those of the Dymola simulation platform, dedicated to the simulation of thermodynamic systems. The proposed control system uses as feedback the overall energy consumption of the system, composed of chillers and tower fans. It uses the variation of the fan speed setpoint in order to achieve energy efficiency. The extreme search is based on the continuous variation of the system setpoint, sometimes increasing, sometimes decreasing its value, causing an increase or decrease in the total energy consumption, after a given time interval. In this way, convergence on the point of lowest global energy consumption is obtained by monitoring the gradient of electrical energy consumed in a given time interval. In addition to the main control loop, there are two other proportional-integral control loops internal to the main one, in order to ensure that chiller and condensing water temperatures are within the operating limits of the chiller. The results show energy savings of up to 5.7%.

In [9], a control strategy defined as OAT (optimum approach temperature) is proposed for the electrical power optimization of the cooling tower. Therein, the term "approach" represents the difference, at a given instant, between the condensing water temperature and the wet bulb temperature. This temperature difference has a direct impact to the effectiveness of the cooling tower. The smaller the cooling tower approach, the greater its effectiveness, i.e., the greater the efficiency of the thermal exchange performed. However, in terms of the system's power consumption, the coolest approach does not always represent the best operating condition, as this condition demands greater consumption by the cooling tower fans. The OAT algorithm is an optimization strategy that can only be applied to cooling towers.

The strategy using the OAT algorithm, unlike the one proposed in this work, is not based on a thermodynamic modeling of the process, nor on that of the equipment that composes it. Instead, it is based only on the analysis of data obtained in the field. Therefore, the definition of the best setpoint for the temperature of the condensing water depends exclusively on the quantity and quality of the obtained database. It is not always possible to obtain a significant amount of data for modeling a system that is composed of different pieces of equipment. In this work, the approach based on thermodynamic modeling of the system equipment is preferred. It is noteworthy to emphasize that the aforementioned strategies are not optimization approaches.

3.2. Evolutionary Strategies

The work reported in [10] presents an optimal control strategy for a chiller-based cooling system in order to improve the overall energy efficiency. In this case, the precision of the modeling of the system's equipment is guaranteed through the online updating of its parameters, using the recursive least-squares method. A genetic algorithm is used as a global optimization tool, whose evaluation function minimizes the global consumption of energy demanded by the chillers, fans, and condensed water pumps. The genetic algorithm chromosome represents of the temperatures of chilled water, condensed water, and water in the heat exchanger. During system implementation, the dynamic simulation environment TRNSYS was used. The condensing water pumps and tower fans have frequency converters, allowing them to vary their speeds to obtain the desired energy savings. The results indicate daily savings of between 0.73% and 2.55% when compared to the operation with conventional adjustments, i.e., with the equipment operating with a fixed speed equal to the nominal reference.

In [11], an optimization strategy based on a modified genetic algorithm is presented. In this case, the system is composed of chillers, condenser water pumps, and cooling tower fans. The constraints of the problem are the system's components and their interactions. In this work, the condensing water pumps and the tower fans are activated through frequency converters, allowing the variation of their speeds. The variables that are represented in the chromosome of the genetic algorithm are: the numbers of chillers, pumps, and tower fans that must remain in operation, and the input and output flows and temperatures of the condensed water circuit. The genetic algorithm's evaluation function considers the linear combination of the system's constraints, using penalty factors for each constraint that must be verified. The objective is to minimize the energy consumption of the overall system. The main differences from the classical genetic algorithm are the fact that the search space is restricted to a pre-defined region around the result of the previous optimization, and the use of the elitism technique, allowing the best results obtained to be kept for the next generation. The proposed system was tested in a pilot HVAC plant (heating, ventilating, and air conditioning), showing promising results. Three scenarios were compared: (i) flows of condensing water and air in the constant tower fans; (ii) constant condensing water flow and air flow in the variable tower fans; and (iii) variable water and air flows in the tower. The third scenario showed the best results. The results show savings of up to 30%, and for values close to the nominal thermal load, that is, part load ratio (PLR) close to 1, savings of 10% were obtained.

Regarding the optimization strategies in [10,11] and the one proposed in this work, there is a crucial difference, which is the multi-objective approach. The optimization is done not only in terms of energy efficiency, but also in terms of the effectiveness of the thermal exchange performed in the cooling tower. Thus, the multi-objective approach, in contrast with the previous ones, allows selecting the best solution that offers the best trade-off between energy savings and tower effectiveness reduction. The operation of a cooling tower outside the limits of optimal effectiveness causes a reduction in its performance and an increase in the consumption of water in the system, because a poorly imposed air flow by the fans causes an increase in water loss by a dripping process in the tower filling. Otherwise, too much air flow in relation to the water flow causes an increase in the volume of water that is lost by the dragging process. Therefore, in this work, the proposed multi-objective approach provides a significant contribution compared to the existing related works. In addition to the system's overall electrical energy savings, it also considers the effectiveness of the cooling tower.

3.3. Learning Strategies

In [12], artificial neural networks and genetic algorithms are combined for the optimal control of a cooling system based on a LiBr absorption chiller [13]. In this case, the neural network is used for modeling the system, and the learning process of the network is carried out with a database collected from the system. The genetic algorithm is used as a global

optimization tool, whose evaluation function considers the total energy cost for operating the system, which is composed only of chillers, cooling towers, and condensing water pumps. Minimum and maximum limits for refrigerated water and condensing temperature are imposed. The neural network used is a common multilayer perceptron (MLP) [14] with two layers. The main layer is composed of five neurons, and the hidden layer, nine neurons. The implemented neural network has five inputs and four outputs. The inputs are the chiller thermal load, the chilled water circuit mass flow, the condensing water circuit mass flow, the chilled water's temperature, and the condensing water temperature. The outputs are the chiller's fuel consumption rate, the chiller's coefficient of performance (COP), the power consumption of the water circuit pump, and that of the condensed circuit.

The training process of the neural network used in [12] demands a large amount of processing data, mainly due to the considerable number of inputs and outputs. As the focus of the work proposed in this paper is the optimization of the cooling system and not its modeling, it is established that the behavior of the system is represented by its thermodynamic modeling, differently from the work reported in [12]. In the proposed work, a non-linear regression algorithm is used to obtain the coefficients of the thermodynamic modeling of the cooling tower and the power consumption of the compression chiller. Moreover, in [12], the power consumption of the cooling tower fans is not considered in the neural network modeling, as this amounts to approximately 1% of the total system requirements. Note that this represents a case apart, as this proportion is not always valid. In the case of the work proposed in this paper, this proportion is 11.3%.

3.4. Swarming Strategy

In [15], an implementation using a hybrid optimization algorithm is presented, through the combination of two distinct techniques, to obtain the lowest energy consumption of a chiller-based chilled water system. The system was composed of a chiller, pumps, and a cooling tower. To simulate the process, the EnergyPlus simulator was used, which features customized tools for thermodynamic equipment and processes. The algorithms used in the optimization were particle swarm optimization (PSO) and the Hooke–Jeeves algorithm. Optimization was achieved by varying the temperature setpoints of the condensing and chilled water circuits. First, the PSO algorithm was applied, and the search space was defined based on the allowable range for temperatures. The result obtained by the PSO was then used as input in the Hooke–Jeeves algorithm, which had a smaller search space, established based on a pre-defined interval inferred from the value obtained by the PSO algorithm. The results indicate total energy savings of 9.4% in summer and 11.1% in winter, compared to running without the optimization support. These results were achieved by updating the setpoints every hour. The results also show that as the frequency of updating the temperature setpoints obtained through the optimization tool increases, the results improve in terms of energy savings. To obtain the results, a database corresponding to four days of operation in summer and winter was used.

In [16], a proposal similar to the one reported in [17] is reported, with a system composed of the same equipment. In this work, MLP-type artificial neural networks are also used to survey the system's predictive model. However, four neural networks are employed: one for predicting the energy consumption of each of the following: the chiller, pump, fan, and reheating device. The four neural networks have nine inputs and a single output. The optimization algorithm used is particle swarm optimization (PSO), which obtains the ambient temperature and static air pressure values that best meet the defined cost function. The optimization system continuously adjusts the temperature and static air pressure setpoints within the established limits for the search space regarding each variable. The objective consists of minimizing energy consumption throughout the day. The optimization results show a reduction in the energy consumption of the HVAC system by around 7%.

Among the approaches presented in [15,16], only the one reported in [17] presents a multi-objective approach to solve the problem involving the search for the best operational

condition of the chiller-based cooling system. However, the second considered objective is the maximization of thermal comfort, unlike the implementation proposed in this work. As the second objective, we consider the maximization of the effectiveness of the thermal exchange carried out in the cooling tower. Furthermore, the cooling system proposed in [17] is not based on a cooling tower and chiller, but only on a chiller with coupled fan. In addition, from the point of view of implementation, the use of neural networks presents a disadvantage in relation to thermodynamic modeling of the cooling system. For this purpose, it is necessary to have a considerably large database to be able to achieve correct and adequate modeling and training of the network. Note that this is mandatory in order to guarantee a good result not only in terms of optimization, but also in terms of operational safety. On the other hand, thermodynamic modeling, as adopted in the proposed work, needs system data only in order to evaluate the designed model and to carry out small adjustments of that model. Therefore, the latter approach is more reliable in terms of results and operational safety. Furthermore, the training stage and selection of the variables and topology of the network is a slow process that depends on the quantity and quality of the data obtained.

4. Problem Formalization

The cooling system to be optimized is composed of chillers and cooling towers. This configuration is commonly used in commercial buildings and industrial facilities to ensure thermal comfort of the transiting people and adequate equipment cooling. The configuration of the cooling system considered in this work is presented in Figure 3. It includes two cooling towers with a capacity of 2500 TR each. The TR unit stands for ton of refrigeration and is commonly used for refrigeration systems, where 1 TR $\approx$ 3.5168 kW corresponds to the cooling power that provides the amount of heat needed to melt a ton of ice in 24 h.

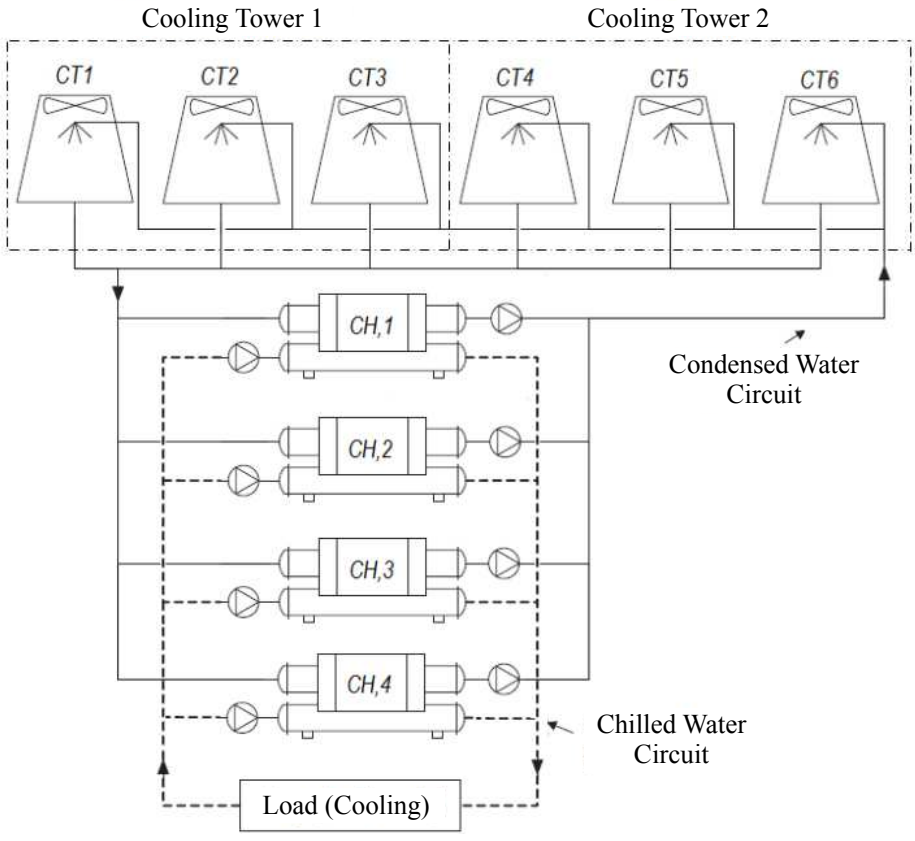

Figure 3. Refrigeration system's configuration.

Each cooling tower is composed of three elementary cells, as shown in Figure 3, where a fan operates in each cell whose electric motor has a nominal power of 30 HP. Powers of electric motors are generally indicated in HP, which stands for horse power, where 1 HP is equivalent to 735.5 W. The two cooling towers must meet the thermal demand of four chillers of 1000 TR. The rated power of the electric motor that drives the compressor of each chiller is 586 kW. The rated power of the condensed water lift pumps and chilled water circulation pumps motors is 120 HP. Among these components, only the fans of the cooling towers allow speed variation, through the use of frequency converters, while the others always remain operating at a fixed speed and equal to the nominal one.

Generally, the daily thermal load is met with the use of two chillers. The third chiller is available for sporadic conditions of additional thermal load, such as on days when the number of people and equipment exceed normal operating values or on extremely hot days in summer. The fourth chiller would only operate in a situation of operational rotation, a situation in which there is periodic alternation between the chillers in operation. This is so in order to avoid excessive wear of certain equipment to the detriment of other equipment, or in case of failure of one of the other pieces. Therefore, the situation of operation with two chillers is the most common for the cooling system to be optimized, as considered in this work.

The compression chillers used were from the manufacturer York®, model YKLKLLH9-CZFS, with rated voltage of 4.16 kV, thermal capacity of 3517 kW (1000 TR), rated electrical power of the compressor motor of 586 kW, and use R-134A refrigerant gas. The complete specification of chillers can be found in [18].

The cooling towers were of the dry type, with mechanical draft induced in counter-current. Each cell in the towers contains two layers of infill. The tower's manufacturer was Alpina®, model 3-CPL-63/79/50-ASP. The nominal design conditions can be obtained from [19]. Each composing cell of the cooling towers can be represented as an independent unit where heat exchange takes place. The cells are operationally independent from each other. Each cell includes the following individual systems: a hot water distribution system, water spray nozzles, filling, and a fan, which induces a current of external air into the cell.

In the proposed problem, the optimization of the heat exchange efficiency that occurs within each cell is obtained after determining the optimal relationship between the water and air flows. Each lift pump of condensed water operates with a nominal flow of 505 m^3/h. As defined in [18], the nominal flow in the condenser of the chiller corresponds to 496.8 m^3/h. Thus, the number of condensed water lift pumps must correspond, in the case under study, to the number of chillers, in order to meet the nominal flow in the chiller condenser. The decision on how many tower cells should operate at a given moment depends on how many condensed water lift pumps are in operation, so that the input flows into the tower cells are always within their operating margins. The minimum and maximum operational limits for the input flow in the cooling tower cells correspond, respectively, to 30% below and 20% above the nominal input flow. The latter is 404 m^3/h, as indicated in [19]. Therefore, the input flow into each cell of the cooling towers must remain between 282.8 and 484.8 m^3/h.

In the case under study, the number of condensation water lift pumps in operation must be equal to the number of chillers in operation. Thus, it can be said that the total number of cells in operation in the cooling towers can also be obtained based on the number of chillers in operation. Thus, Table 1 indicates the various possible scenarios contemplating situations with up to 4 operating chillers, where the number of condensed water lift pumps and the total number of cells can be observed of the cooling towers that must operate in each situation, in order to maintain the minimum and maximum flow limits for the chillers and the tower cells. In Table 1, the indicated flows are in m^3/h. The configurations for which the impossibility of carrying out real tests was verified are marked with "−", since, as predicted for the respective theoretical value, the real flow regarding these cases would be outside the operational limits of the cells, which could damage the tower components. It is noted that the predicted theoretical flows are dependent on the

sum of the flow rates of the lifting pumps. However, it is known that the association of centrifugal pumps plugged in parallel provides a non-linear reduction in the individual flow rate of the pumps. Additionally, as pumps are added in parallel, each pump will always operate with a flow rate lower than the one verified previously. The purpose of this approximation is to obtain an estimate to help verify whether the actual measurement of the inlet flows in the tower cells is carried out to avoid improper operation and/or possible damage. Thus, as indicated in Table 1, in the case under study, for a total of n chillers in operation, we use a total of $n + 1$ tower operating cells. This guarantees that the cells will always operate within their operational limits of inlet flow.

Table 1. Inlet flows into the cooling tower cells in m^3/h.

#Chillers	#Pumps	#Cells (Theoretical)					#Cells (Real)				
		1	**2**	**3**	**4**	**5**	**1**	**2**	**3**	**4**	**5**
1	1	505.00	252.50	168.30	–	–	550,00	280.00	170.00	–	–
2	2	1010.0	505.00	336.70	252.50	–	–	485.00	330.00	–	–
3	3	1515.0	775.50	505.00	378.75	303.00	–	–	410.00	320.00	–
4	4	2020.0	1010.0	673.30	505.00	404.00	–	–	–	–	380.00

The operation of cooling towers in conjunction with chillers of compression demands a large amount of electrical energy. A building with a cooling system as configured in this work would require overall consumption of approximately 60% of the total electrical energy spent. The temperature of the cooled condensed water, which leaves the towers and goes to the chiller condenser, directly influences its energy consumption according to a non-linear model composed of several variables. This will be detailed later in Section 5, when the system's model is presented. Generally, higher temperatures for the condensed water cause an increase in energy consumption in the chiller. To obtain lower temperatures for the condensed water, the speed of the cooling tower fans must be increased. This is due to the fact that in the case under study, the condensed water lift pumps operate at a fixed speed. However, increasing the speed of the tower fans also causes an increase in energy consumption.

Currently, most cooling towers are operated in order to always provide the lowest possible temperature for the condensed water. However, it is known that this minimal feasible temperature to be reached is limited to the installed wet bulb temperature. Therefore, operationally, a situation can be reached where there is no advantage in keeping the fans operating at their rated speed, since there is a speed adjustment (setpoint) below the rated speed which is adequate to reach the wet bulb temperature. Thus, the challenge here is to discover this setpoint speed for the tower fans. Furthermore, according to [9], the strategy for controlling the tower outlet temperature in order to always attempt to reach the local instantaneous wet bulb temperature is not always the best option. There are situations in which, from a certain value of condensed water temperature, its reduction no longer influences the consumption of the chiller, tereby causing unnecessary consumption of energy in the tower fans. Thus, only with appropriate, precise modeling of the equipment would it be possible to estimate the influences of process variables on the energy consumption of each piece of equipment (chillers and cooling tower fans). From the modeling and implementation via an optimization process and decision support system, the best operational adjustments (setpoints) for the cooling system can be determined, in order to achieve high energy efficiency levels.

In addition to the efficiency related to the electrical energy consumption of the cooling system, another aspect to be taken into account is the efficiency related to the thermal exchange that occurs in the cooling tower. This is defined as the system's effectiveness. Energy efficiency is as important as the tower's effectiveness, because the latter is associated with the water consumption of the cooling system. Moreover, the operation of a

cooling tower is conditioned to a normal loss of water, which is estimated based on the expected system's effectiveness and dimensions of the cooling tower. If it operates with an effectiveness below that estimated, it will consume more water, and consequently cause an increase in energy consumption.

The effectiveness of the cooling tower is defined as its operational efficiency, and is related to the efficiency of the heat exchange between the hot water coming from the process and the air mass induced in the tower through the counter-current, via fans. This efficiency is influenced by several factors, which are explained in the modeling of the cooling tower, in Section 5. Note that the actual models are presented and validated in [4] for the cooling tower and in [5] for the chiller. Among the factors that influence the effectiveness of the tower, we have the relationship between the water and air flows inside the tower and climatic factors, defined by external and wet bulb temperatures. In this work, the water flow that reaches the tower cells only varies as a function of the number of pumps that are in operation, i.e., as a function of the number of operating chillers. On the other hand, the air flow in each cell can vary continuously through the variation of the fan speed. The external temperature influences the thermal load to be served by the chillers, and the wet bulb temperature influences the efficiency of the thermal exchange of the tower, as it represents the lowest possible outlet temperature to be reached. Thus, this work explores multi-objective optimization in order to solve the problem composed of the following conflicting objectives:

- Maximizing the effectiveness of the cooling tower;
- Minimizing the overall electrical energy consumption of the refrigeration system.

To this end, the process variables were collected in the field from the instrumentation already installed in the cooling towers. Local weather conditions were provided by a weather station installed and integrated into the cooling system. Thus, based on the process data provided by the existing supervisory control and data acquisition (SCADA) system, the following variables were provided as inputs to the optimization system: the number of chillers that are in operation; the temperature of the hot water reaching the cooling tower; the wet bulb temperature on site; the flow of water that reaches the cooling tower; the water flow that leaves each chiller.

In this work, two different models were implemented for the compression chiller, defined, respectively, by the factors Z_{CAP_1} and Z_{CAP_2}, used in the modeling of its energy consumption. The implementation of the two models was performed in order to test two different scenarios for the optimization system, where different search variables can be considered in the implementation of the optimization algorithms. This allowed us to compare the two models, in order to identify the best scenario for the application of the proposed multi-objective optimization. Thus, the two scenarios were defined in this work as:

- Modeling based on Z_{CAP_1}: it considers only adjustment in the speed of the tower fans;
- Modeling based on Z_{CAP_2}: it considers adjustments in the speed of the tower fans and adjustments in temperature of the water leaving the chillers.

Thus, considering the application of Z_{CAP_1} modeling, the optimization system should provide as output: the optimal setpoint for the speed of the cooling tower fans. Considering the application of Z_{CAP_2} modeling, the optimization system should provide as output: the optimal setpoint for cooling tower fans and that for the temperature of the chilled water provided by the chillers.

When analyzing the defined two scenarios a priori, it is possible to note that the Z_{CAP_1} modeling has the advantage of dealing with only one search variable, whereas the Z_{CAP_2} modeling must deal with two conflicting variables. However, nothing can be said about the influence of this fact on the results to be found. In this study, the objective was to verify the impact of increasing a search variable on the time required for the optimization system to provide the optimal solution. The results of energy savings and tower effectiveness obtained after implementing the optimizations using the two models are compared herein.

According to the aforementioned explanation, it is clear that a cooling tower operates in conjunction with other equipment. Among this equipment, the chillers occasion the highest energy consumption. Since the condensed water and chilled water circulation pumps always operate at a fixed speed, the inclusion of these into the calculation of the overall energy required by the cooling system does not provide any advantage, since the objective is to evaluate the energy efficiency as achieved after application of the optimization algorithms. Thus, only the consumption levels of the chillers and tower fans are considered in the implementation of the proposed energy optimization system.

As a premise for the implementation, it was considered that the optimal output values of the optimization system must be obtained based on the best compromise between the objectives established above, while respecting the operational limits and restrictions defined for the equipment that compose the cooling system. The objective was to obtain, at each predefined interval of one hour, the best setpoint of speed for the tower fans and/or the best setpoint of temperature of the water leaving the chiller, depending on the modeled scenario. The optimization simulations were performed using the two distinct swarm-based multi-objective optimization algorithms, namely, multi-objective particle swarm optimization (MOPSO) [20] and multi-objective TRIBES (MO-TRIBES) [21], in order to compare the results obtained for the two considered models, Z_{CAP_1} and Z_{CAP_2}, using different stopping criteria. Note that a brief explanation of the dynamics of the used optimization algorithms is presented in Section 6.

4.1. Objective Functions

In this work, we envisage two conflicting objective functions to be used to solve the proposed problem: f_1, which evaluates the tower's effectiveness, and f_2, which evaluates the required power of the refrigeration system. f_1 is used to maximize the heat exchange efficiency of the cooling tower and f_2 to minimize the electrical energy consumption of the cooling system, which is composed of the following equipment: cooling tower fans, chiller compressors, and lifting pumps that take the water from the chillers for the cooling tower.

Among the equipment that composes the cooling system considered in this work, only the tower fans allow speed variation, through the use of frequency converters. Lift pumps and chillers operate at a fixed speed, which is equal to the rated speed. Thus, as the condensed water pumps are not influenced by the speed variation of the tower fans, nor by the variation in the temperature of the water passing through the chillers, in the condenser or in the evaporator, the required energy is not be taken into account in the optimization process. Therefore, the optimization is dedicated to the electrical energy demands of the fans and the chillers. Objective function f_1, referring to the efficiency of the heat exchange of the cooling tower, is defined in Equation (1):

$$\text{(maximize)} \quad f_1 = \epsilon_a, \tag{1}$$

wherein ϵ_a represents the effectiveness of the cooling tower. Objective function f_2, referring to the electrical power demanded by the system composed of chillers and cooling tower fans, is defined in Equation (2):

$$\text{(minimize)} \quad f_2 = n_1 P_v + n_2 P_{chiller} \tag{2}$$

where n_1 and n_2 are discrete variables, representing the numbers of fans and chillers that must operate in order to meet the requested thermal demand and the commitment to lower energy consumption, respectively. Moreover, P_v and $P_{chiller}$ represent the electrical power demanded by fans and chillers, respectively. Recall that the number of fans in operation corresponds to the number of tower cells required in order to guarantee its operational limits. After analyzing the values presented in Table 1, and as explained earlier, we must have $n_1 = n_2 + 1$.

Restrictions

For the optimization problem, four operational constraints related to the considered cooling system are required to guarantee correct system operation. These are addressed in the sequel.

The first constraint, g_1, concerns the lowest possible value to be reached by the cooling tower outlet temperature. It cannot be lower than the local instantaneous wet bulb temperature due to saturation of the air leaving the tower after heat transfer and the mass of the hot water that reaches the tower. The wet bulb temperature varies throughout the day and can be calculated as a function of ambient temperature and relative humidity. Therefore, the first restriction is defined as $g_1 : T_{as} \geq T_{BU}$, wherein T_{as} represents the temperature of the water leaving the cooling tower and T_{BU} the wet bulb temperature.

The second constraint, g_2, concerns the operating conditions of the chiller considered in this work. The manufacturer of the chiller established in [22] a restriction regarding the temperature difference between the water inlet and outlet of the condenser. The surge curve of the chiller is shown in Figure 4, where it is possible to observe two operating zones for the chiller: with or without surge. The operation in the surge zone of the chiller compressor causes a series of inconveniences, such as vibrations and load oscillations, generating wear in its mechanisms and undue performance of the electrical protection associated with overload. Furthermore, in this operating condition there is a considerable reduction in the coefficient of performance (COP) of the equipment.

The COP of a chiller represents the relationship between the cooling capacity ($kW_{thermal}$) and the electrical power required ($kW_{electric}$) for its operation. Thus, the chiller should preferably operate in the zone below the surge line, as shown in Figure 4, which represents the maximum admissible limit for the temperature difference between the inlet and outlet of water in the condenser as a function of the chiller load.

Figure 4. Regions with or without surge from the chiller.

Thus, the second restriction is defined as $g_2 : \Delta T_{co} \leq 7,3c_t - 0.3$, with $\Delta T_{co} = T_{ae} - T_{as}$, wherein c_t represents the load factor of the chiller, where $c_t \in [0, 151]$, T_{ae} the temperature of the water that leaves the chiller condenser and leaves towards the cooling tower, and T_{as} the temperature of the water leaving the cooling tower and going towards the condenser inlet. Moreover, from Figure 4, it can be observed that the manufacturer does not recommend operating the chiller with a load below 15%, as indicated in [22]. This imposes the third constraint, which is defined as $g_3 : 15\% \leq c_{t\%} \leq 100\%$.

The fourth restriction concerns the maximum limit of the water inlet temperature in the cooling tower. As indicated in [19], the nominal design temperature of the cooling tower corresponds to 36.4 °C. Therefore, temperatures above this value should be avoided, since

the manufacturer does not guarantee the tower's performance for inlet temperatures that exceed the nominal value. In this way, we define the fourth constraint as $g_4 : T_{ae} \leq 36.4$.

5. System Modeling and Objective Functions

In this section, we give a report on the overall models of the refrigeration system regarding its effectiveness and power consumption. This forms the basis for defining the two objective functions to be optimized using the swarm intelligence-based techniques. Thus, we need to provide a model to compute the system's effectiveness ϵ_a, which defines the first objective function f_1, and a model to compute electrical power demanded by fans P_v and chillers $P_{chiller}$, which define the first objective function f_2.

The models used for the main equipment of the refrigeration systems are based on Merkel's method for evaluating a cooling tower 's performance [23] and Braun's method for evaluating the system's effectiveness [24]. We propose a complete model for a modern refrigeration system based on cooling towers, fans, and chillers. The details of the modeling proposed for the cooling tower with the entailed fans are presented in [4]. The details of the model proposed for the chillers are described in [5]. For self-completeness of the work described herein, we describe the final results obtained, which will help in defining the objective functions of the optimization process. Thus, using the resulting models for the cooling towers, fans, and chillers, we obtain the definitions of the objective functions. The first objective, given by Equation (3), is to maximize the effectiveness of the tower ϵ_a. (For details, see [4].) The second objective, given by Equation (4), is the minimization of the instantaneous global electrical power demanded by the cooling system, based on P_v and $P_{chiller}$. (For details, see [5].) Therefore, the objective functions are:

$$\begin{aligned} \max f_1 = \; & c_0 + c_1\left(\frac{\dot{m}_{air}}{\dot{m}_{water}}\right) + c_2(T_{water_{in}} - T_{BU}) + c_3\left(\frac{\dot{m}_{air}}{\dot{m}_{water}}\right)^2 + c_4(T_{water_{in}} - T_{BU})^2 + \\ & c_5\left(\frac{\dot{m}_{air}}{\dot{m}_{water}}\right)(T_{water_{in}} - T_{BU}); \end{aligned} \tag{3}$$

$$\begin{aligned} \min f_2 = \; & n_1\sqrt{3}V_n I_n \times \left(d_0\left(\frac{\dot{m}_{air}}{\dot{m}_{air_n}}\right)^3 + d_1\left(\frac{\dot{m}_{air}}{\dot{m}_{air_n}}\right)^2 + d_2\frac{\dot{m}_{air}}{\dot{m}_{air_n}} + d_3\right) + \\ & n_2 Q_{chiller_{no}} EIR_{no} Z_{CAP_i}(T_{ae_{co}}, T_{ae_{ev}}) \times Z_{EIR}(T_{ae_{co}}, T_{ae_{ev}}), \end{aligned} \tag{4}$$

wherein factors Z_{CAP_i}, for $i = 1, 2$ and Z_{EIR}, in Equation (4), are defined as in Equations (5)–(7), wherein we have $\Delta T_{ag} = T_{ae_{ev}} - T_{as_{ev}}$ [5]:

$$Z_{CAP_1} = \; b_0 + b_1 T_{ae_{ev}} + b_2 T_{ae_{ev}}^2 + b_3 T_{ae_{co}} + b_4 T_{ae_{co}}^2 + b_5 T_{ae_{ev}} T_{ae_{co}}; \tag{5}$$

$$Z_{CAP_2} = \; b_0 + b_1 \Delta T_{ag} + b_2 \Delta T_{ag}^2 + b_3 T_{ae_{co}} + b_4 T_{ae_{co}}^2 + b_5 \Delta T_{ag}^2 T_{ae_{co}} + b_6 \Delta T_{ag} T_{ae_{co}}^2; \tag{6}$$

$$Z_{EIR} = \; a_0 + a_1 T_{ae_{ev}} + a_2 T_{ae_{ev}}^2 + a_3 T_{ae_{co}} + a_4 T_{ae_{co}}^2 + a_5 T_{ae_{ev}} T_{ae_{co}}. \tag{7}$$

It is noteworthy to recall that all the aforementioned variables are fully defined herein or in the model descriptions of the cooling tower and fans [4] and/or of the chillers [5]. Additionally, recall that the two the usage of either Z_{CAP_1} or Z_{CAP_2} follows from the consideration of two optimization scenarios. This calls for two different models for the chillers. We may consider that the evaporator outlet temperature does not change, and the optimization can only be focused on finding the setpoint regarding the speed of the cooling tower fans, which can be achieved by varying the condensation water temperature. Instead, we may consider that optimization can target finding the setpoints of the temperature of the evaporator outlet water and that of the speed of the fans in the tower. More details on the motivations behind the consideration of these two optimization scenarios can be found in [5].

In the implementations using the Z_{CAP_1} modeling, the search variable is $\dot{m}_{air}$, which represents the mass air flow. It is proportional to the air velocity of the cooling tower fans. In the implementations of the optimizations using the Z_{CAP_2} modeling, there are two search variables: $\dot{m}_{air}$ and $T_{as_{ev}}$; the latter represents the temperature of the chiller's outlet cooled water. Note that the model requires the definition of four sets of coefficients:

$a_0 \ldots a_5$ and $b_0 \ldots b_5$, for Z_{CAP_1}; $b_0 \ldots b_6$ for Z_{CAP_2}; $c_0 \ldots c_5$ and $d_0 \ldots d_3$. These coefficients were obtained via a prior optimization process. We exploited the Levemberg–Marquardt non-linear regression technique [25]. Their values are given in Table 2. Furthermore, recall that the operational limits of the equipment considered in this work are defined and implemented in the optimization system through four restrictions, as defined earlier.

Table 2. Obtained coefficients to model the effectiveness and power required by the refrigeration system.

Z_{EIR}	Value	Z_{CAP_1}	Value	Z_{CAP_2}	Value	ϵ_a	Value	P_v	Value
a_0	−1.0405	b_0	−0.8108	b_0	−0.1177	c_0	+0.0262	d_0	+0.7931
a_1	+0.1379	b_1	−0.0838	b_1	+0.3381	c_1	+0.4935	d_1	+0.0330
a_2	−0.0090	b_2	+0.0133	b_2	−0.0513	c_2	+0.14350	d_2	+0.0557
a_3	+0.0840	b_3	+0.0997	b_3	−0.0276	c_3	−0.0289	d_3	+0.0039
a_4	−0.0022	b_4	−0.0012	b_4	+0.0022	c_4	−0.0129		
a_5	+0.0033	b_5	−0.0032	b_5	+0.0030	c_5	−0.0533		
				b_6	−0.0006479				

6. Multi-Objective Swarm Intelligence Algorithms

In this section, we briefly define the main concepts regarding multi-objective optimization and swarm intelligence. Then, we motivate the choice of the applied algorithms. Then follow their descriptions and related issues.

6.1. Multi-Objective Optimization Concepts

Multi-objective optimization can be defined as a process that seeks to find the values for decision variables that satisfy pre-established constraints and optimize a given application, represented by two or more conflicting objective functions. Due to the latter fact, the concept of finding an optimum is no longer a simple solution and becomes an attempt to obtain good compromises between the different objectives, i.e., trade-offs. Thus, there is not a single optimal solution (or the best solution) for a given problem; instead, there is a set of solutions that represent the best trade-offs, considering the considered objectives. If there is a set of possible solutions that meet the pre-established restrictions and that have good compromises between the defined objectives, it is necessary that a decision maker, based on preferential information, is able to select the solution to be adopted.

A multi-objective optimization problem (MOP) is defined by a set of n decision variables, a set of k objective functions, and m constraints. Assuming that the objective of optimization is to maximize the objective functions of Equation (8) [26]:

$$y = f(x) = (f_1(x), f_2(x), \ldots, f_k(x)), \tag{8}$$

and that the restrictions are defined as in Equation (9):

$$g(x) = (g_1(x), g_2(x), \ldots, g_m(x)) \leq 0, \tag{9}$$

wherein we have the decision vector and the objective vector defined as in Equation (10):

$$x = [x_1, x_2, \ldots, x_n] \in X, \quad y = [y_1, y_2, \ldots, y_n] \in Y, \tag{10}$$

where X is the decision space and Y is the objective space, the constraints g determine the set of feasible solutions. Using the Pareto dominance concept, two feasible solutions a and b of a multi-objective minimization are compared and ranked, according to Equation (11), for $i = 1, \ldots, n$ goals:

$$\begin{array}{l} \text{if } f_i(a) > f_i(b), \text{ then}, a \succ b \quad (a \textit{ dominates } b); \\ \text{if } f_i(a) \geq f_i(b), \text{and } \exists\, j \in [1,n]\,;\, f_j(a) = f_j(b), \text{ then}, a \succeq b \,(a \textit{ dominates loosely } b); \\ \text{if } f_i(a) \leq f_i(b), \text{and } \exists\, j \in [1,n]\,;\, f_j(a) < f_j(b), \text{ then}, a \sim b \,(a \text{ is } \textit{indifferent} \text{ to } b). \end{array} \tag{11}$$

6.2. Swam Intelligence Concepts

Swarm intelligence is a computational intelligence paradigm that is dedicated to studying the collective behavior of decentralized multi-agent systems. Such agents, making an analogy to nature, are individuals belonging to a swarm. They interact locally with each other and also with the environment to generate solutions for a given problem [27]. In models based on swarm intelligence, the swarm concept can be applied in spaces where collisions between agents, i.e., occupation of the same space simultaneously, do not have major negative impacts, as could happen in the natural world. In this case, a collision can be understood as a convergence, keeping a probabilistic meaning [28]. According to [29,30], there are five basic principles of swarm intelligence: (i) proximity—the swarm must consider simple space and have a time reference; (ii) quality—the swarm needs to respond to qualitative factors of the environment; (iii) response diversity—the swarm cannot commit to exploring very specific paths; (iv) stability—the swarm should not change its behavior with each change in the environment; (v) adaptation—the swarm must be able to change its behavior whenever the computational cost is prohibitive.

According to the literature review carried out on multi-objective particle swarm algorithms, we found out that MOPSO is the algorithm with the highest number of verified applications, both in terms of proposed improvements in order to guarantee convergence and diversity and in relation to practical applications, mainly because it is based on the PSO algorithm. Regarding mono-objective optimization, the results achieved by TRIBES, modified TRIBES (MTribes), and PSO algorithms indicate the superiority of MTribes and TRIBES compared to PSO, mainly with regard to the number of iterations necessary for the problem resolution [31]. For these, reasons, we chose to apply MOPSO and MO-TRIBES and compare their performances for the application. Note that in a previous work, we applied evolutionary multi-objective algorithms [3].

6.3. MOPSO

The multi-objective algorithm based on a particle swarm was initially introduced in [32], and from this first approach, other versions of MOPSO emerged [20,33–35]. One of the first versions of the algorithm was proposed in [34], where the use of lexicographical ordering was considered as a technique for solving multi-objective problems. In [36], the use of the external file was introduced. It is characterized as a memory extension, and aims to store the non-dominated solutions obtained, reducing the convergence time of the algorithm. In [20], the MOPSO algorithm was proposed. It considers the use of an external file to store the best experience of each particle. The search space is divided into a pre-defined number of hypercubes. The density of each hypercube based on the solutions stored in external files is evaluated after each iteration. In this way, each hypercube receives a score. The guide for each particle, also known as the particle's best (*pbest*) is chosen based on the roulette technique, using the values stored in the external file. The global best (*gbest*) is chosen after determining the hypercube that has the best score, based on the lowest density. A particle in this hypercube is chosen randomly. In [35], a version of MOPSO that uses Pareto dominance to determine the search direction of a particle was proposed. The algorithm uses a file to store the non-dominated solutions and adopts a mutation operator to generate new possible solutions within the unexplored search space to maintain the swarm diversity.

Finding the solution of a multi-objective problem is about finding the set of solutions that optimize the problem by achieving good compromises between all considered objectives. This set known as the Pareto front. Thus, in order to allow the particles to have the capacity to produce diverse non-dominated solutions in each iteration, one needs to guarantee the proper implementation of the following aspects [20]: (i) selecting leading particles in order to give preference to non-dominated solutions and ignore dominated ones; (ii) storing the non-dominated solutions found during the process after comparison with the non-dominated solutions obtained previously, in order to guarantee good diversity

within the solutions that compose the Pareto front; (iii) maintaining the diversity of particles in the swarm, in order to avoid convergence to local minima/maxima.

The MOPSO algorithm performs several iterations until a predetermined stopping criterion is met. At the end of the execution, a set of Pareto-optimal solutions is returned. These are stored in the final R. Algorithm 1 summarizes the steps of the MOPSO, where i_{max} represents the number of particles in the swarm and $iter_{max}$, the maximum number of iterations.

Algorithm 1 Main steps of MOPSO.

Require: $iter_{max}$
Ensure: R
for $i := 0 \rightarrow i_{max}$ **do**
  generate initial random position $x[i]$ of swarm S
end for
for $i := 0 \rightarrow i_{max}$ **do**
  generate initial random speed $v[i]$
end for
evaluate *fitness* of each particle in S; store x_i of the non-dominated particles of S in R
generate hypercubes and allocates the particles in R
for $i := 0 \rightarrow i_{max}$ **do**
  $pbest[i] := x[i]$
end for
while $iter < iter_{max}$ **do**
  for $i := 0 \rightarrow i_{max}$ **do**
    select h using *fitness sharing*; $g[i] := random(R(h))$
    $v[i] := wv[i] + c_1 r_1(pbest[i] - x[i]) + c_2 r_2(g - x[i])$; $x[i] := x[i] + v[i]$
    evaluate $fitness := f(x_i)$ of each particle in S;
    store x_i of the non-dominated particles of S in R
    if E is full when trying to insert the non-dominated vectors of S **then**
      Apply *adaptive grid*
    end if
    if $x[i]$ dominate $pbest_i$ **then**
      $pbest_i := x[i]$
    end if
    $iter := iter + 1$
  end for
end while

Initially, the MOPSO algorithm randomly generates the swarm S, obtaining the initial values of position x_i and velocity v_i of the particles. Then, file R is populated using the non-dominated solutions of P. At the end of the iterations, R should provide the Pareto front for the multi-objective optimization problem. The choice of the guide g_i, at each iteration, for each particle i, is performed as follows: File R is divided into several hypercubes (h_1 , h_2, …, h_n), where n represents the number of hypercubes, which is set by the user. It is resized with each iteration as the coordinates of the new non-dominated particles stored in R extrapolate the limits of the problem's search space. Hypercubes h that contain more than one particle receive an evaluation of fitness, which corresponds to the value of a constant $a > 1$ divided by the number of particles in the considered hypercube. In [35], $a = 10$ is suggested. Thus, the fitness of the hypercubes that have a greater number of particles is reduced. This strategy was seen as a form of fitness sharing in [37]. Then, the roulette selection criterion is applied based on the fitness values obtained for the hypercubes, so as to choose that from which the guide g_i for i will be taken. Once the hypercube is chosen, one of its particles i_h is randomly selected. The particles of swarm S are evaluated through the objective functions, where $pbest_{ij}$ works as a personal memory for each particle i, considering the dimensions j of the search space. The criterion for choosing the best individual position $pbest_i$ that will be

kept by particle i consists of applying Pareto dominance to each new position of the particle (that is, in each iteration), always keeping the non-dominated solution. If the compared particles are indifferent, the choice is done randomly. There are several ways to select the local guide *pbest* from the R, impacting the convergence and diversity of solutions [34,38]. The updating of the velocities and positions of each particle i is performed by Equation (12):

$$v_{ij}^{t+1} = wv_{ij}^{t} + c_1 r_{1j}^{t}(y_{ij}^{t} - x_{ij}^{t}) + c_2 r_{2j}^{t}(y_{ij}^{t} - x_{ij}^{t}). \quad (12)$$

The positions of all particles are then updated by $x_{ij}^{t+1} = x_{ij}^{t} + v_{ij}^{t+1}$. Thereafter, a random value R_T known as the *turbulence factor* is added to the current position of each particle using $x_{ij}^{t+1} = x_{ij}^{t} + R_T x_{ij}^{t+1}$, wherein $R_T \in [-1, 1]$.

The turbulence factor has a similar function to a mutation operator in genetic algorithms. Its objective is to guarantee diversity, preventing the precocious convergence to local minima or maxima. In this case, the mutation operator is not only applied to the swarm particles, but also to the limits of each search variable of the problem to be solved. Throughout the iterations, file R is updated using the following strategy: Upon each iteration, the non-dominated solutions of swarm S are compared to the non-dominated solutions already stored in R. Note that as R initially is empty, current solutions are all accepted. In the following iterations, if the new non-dominated solutions in S dominate the solutions already stored in R, the latter will be replaced by the former. Non-dominated solutions of S that are dominated by any solution in R are discarded. Finally, when R reaches its maximum limit, a routine called *adaptive grid* is applied. The approach used in MOPSO to adjust the external file R is a variation of the adaptive grid proposed in [39]. Thus, it is necessary for the user to provide the number of subdivisions of the grid. The main advantage of using an adaptive grid is its low computational cost when compared the niche-based technique [39], except when the frequency of grid adaptation is high. As indicated in [35], constraints in MOPSO are implemented as follows: whenever two individuals are compared, their constraints are checked. If both are feasible, the dominance criterion is applied to decide which is the winner. If one of them is feasible and the other non-feasible, the feasible dominates. If both are non-feasible, the one with the lowest content associated with the constraint violation dominates.

6.4. MO-TRIBES

TRIBES was idealized and named by Maurice Clerc as a parameter-free and completely adaptive particle swarm optimizer for application in real heterogeneous problems [40]. Heterogeneous problems are multidimensional problems where some dimensions are discrete and others are continuous. When the search space is not a hypercube but a hyper-parallelogram, and when the problem is heterogeneous, MO-TRIBES, which is the multi-objective version of TRIBES, can provide excellent performance. MO-TRIBES was proposed in [21]. In this algorithm, the swarm is divided into tribes, each tribe having a certain amount of particles, which aim to explore the search space investigating for positions that allow obtaining the Pareto front.

MO-TRIBES is inspired by the behavior of tribes looking for good regions. Each tribe is treated as a subswarm [40]. In this way, the swarm is divided into tribes of particles; each tribe acts as an independent swarm, exploring a specific region in particular [41]. Each individual belongs to only one tribe. Each particle moves in a multidimensional search space, where each dimension represents a search variable of the problem. All tribes exchange information about the regions they are exploring. An informant of a given particle i is a particle j that can pass information to i. Generally, this information includes the best position ever found by j and the quality of that position. Therefore, the informant j influences the behavior of particle i. In a tribe, each particle has a cognitive memory p_i, and also receives information from the other particles of its tribe, called internal informants. The best particle of a tribe is called *shaman*, which operates as an external informer with the other tribes. The set of informants of a particle i is called i-group and

contains all particles of its tribe, but is not limited to them. The social memory p_g of a given particle is passed by the informant that represents a non-dominated solution. Mainly, MO-TRIBES proceeds as follows: at the beginning, the swarm is composed of a single particle; according to the behavior of the tribes, particles are added or removed; according to the performances of the particles, their movement strategies are adapted [42,43].

A particle can be classified as *good*, *excellent*, or *neutral*. A particle is said to be *good* when its last move provided an improvement to the tribe ($f(x_i) < f(p_i)$). If the last two moves provided improvements for the tribe (i.e., $f(x_i)^t < f(p_i)^1 \wedge f(x_i)^{t+1} < f(p_i)^2$), the particle is rated *excellent*. If the last move did not improve the tribe (i.e., $f(x_i) \geq f(p_i)$), it is classified as *neutral*. Given a tribe T, the *worst* particle i will be the one where $\forall x_j \neq x_i, f(x_j) \leq f(x_i)$. The best particle i of the tribe T will be the one where $\forall x_j \neq x_i, f(x_j) \geq f(x_i)$.

A tribe can be classified as *good* or *bad* depending on the number of good/bad particles. The more good particles a tribe has, the better it will be. The classification criterion is established through an algorithm that generates a random number between 1 and n_T, where n_T represents the total number of particles in the tribe. It then checks if this random number obtained is greater than or equal to G_T, where G_T represents the number of good particles in the tribe. If so, the tribe is classified as good; otherwise, it is classified as bad. Algorithm 2 presents MO-TRIBES's main steps. Therein, function $f_k(x_i)$ represents the value of the objective function for each objective k, obtained from position x_i of a particle i, NL the number of links of computed information during the last adaptive structure adjustment, and n the number of iterations since the last adaptive restructuring occurred.

Algorithm 2 Main steps of MO-TRIBES.

Require: $iter_{max}$
Ensure: $file$
compute the file size; initialize first particle 0 randomly; evaluate *fitness* := $f_k(x_0)$
$g_0 := p_0 := x_0$; $i_max := 0$; store particle 0 in file; $iter := 0$
repeat
  for $i := 0 \rightarrow i_{max}$ **do**
    determine the situation of particle i; choose movement strategy
    update x_i; evaluate *fitness* := $f_k(x_i)$; update p_i; update g_i; update file
  end for
  if $n > NL$ **then**
    determine the quality of the tribes; apply swarm adaptive structure adjustment;
    add/remove particles from tribes according to tribe quality; update i_max
    re-structure *links*; evaluate the possibility of restarting the swarm
    if $nDomPrev = 0$ **then**
      restart the process; update the file size
    end if
    compute n
  end if
  $iter := iter + 1$
until $iter = iter_{max}$

After a predefined number of iterations, the tribes go through a process called *adaptive structure adjustment*. For each existing tribe, this process allows one to decide whether a particle should be added/removed to/from the tribe. Good tribes should discard the worst particle, and bad ones should receive a new particle, in order to increase the possibility of improving the tribe's performance. Removing the worst particle from a good tribe constitutes an opportunity to reduce the number of fitness evaluations without hindering the tribe's performance [42] process. In the case of a single-particle tribe, it will only be eliminated if one of the informants has a better quality, so as not to lose the information of the particle, if eliminated. The elimination of a particle implies the redistribution of particle's links to maintain the tribe's structure. In general, the tribe best particle takes over

all the eliminated particle links. In the case of a single-particle tribe, the links are assumed by the best external informant particle of the eliminated particle.

Bad tribes need information, and therefore particles to scan the search space. In this way, the respective shaman of each bad tribe generates two new particles and becomes its informant, establishing a link with the other tribes through it. One of the particles is positioned anywhere in the search space, and the other in a very restricted region. These particles are called, respectively, the free particle and confined particle. These will form a new tribe which establishes a link with the shaman of the tribe that generated them. This strategy should allow exploring new positions in the search space. A free particle is randomly generated from a uniform distribution, which is also randomly obtained based on three possible locations. Considering the search space being a hypercube, these three possible locations are: anywhere in the search space; on either side of the search space; at one of the vertices of the search space [42]. Regarding the confined particle generation, let i_s be the shaman of the tribe generating a confined particles, x_s its best memorized position, i_g its best informant, and x_g the best position its best informant has memorized. The confined particle will be generated randomly in the space represented by the hypersphere centered on x_g and with radius of $||x_g - x_s||$.

The adaptive structure adjustment of the swarm should not be carried out after each iteration, because several iterations are necessary for the changes made to the structure to propagate through the swarm and for it to become possible to evaluate their impacts. Theoretically, after each adaptation, the diameter of the interconnection graph between the different particles of the swarm should be calculated, considering only the pairs of particles that belong to different tribes. For this purpose, it would be necessary to calculate the shortest path connecting each pair of particles belonging to different tribes. The longest of the shortest paths thus calculated would be an estimate of the number of iterations needed to be sure that the information carried by a given particle has been transmitted to the rest of the swarm. However, this computation is very computationally intensive. In order to simplify this computation, it would be enough to perform an adaptation step every $L/2$ iterations, where L represents the overall number of links between the swarm particles.

The particle movement strategy is based on its recent past. There are three possibilities depending on the particle performance, which can be either deterioration, maintenance of the status quo, or improvement, symbolized, respectively, by $-$, $=$, and $+$. Since a particle's past is represented by its last two performance evaluations, the algorithm must distinguish nine possible situations that represent a particle's past. The possible situations for the past of a particle are distributed in three groups, where, for each one, a displacement strategy is defined whose details can be found in [42]. Regarding particle movement strategy, MO-TRIBES adds a kind of *natural selection* between strategies. A good particle keeps the same strategy with a probability of 0.5. Unlike classic TRIBES, MO-TRIBES implements two ways to improve diversity in a bad tribe: either generating a new particle or modifying the best position of an existing particle. However, this procedure is only performed for the dimension with the smallest discrepancy considering the entire tribe. Moreover, MO-TRIBES uses the Pareto concept. To do so, it keeps a file always updated with non-dominated solutions obtained throughout the iterations. The solutions are compared based on the criterion of dominance, where the new non-dominated solutions obtained after each iteration replace those already in the file, which are dominated by them.

The size of file is important, and it should be limited, as the update process becomes more complex as its size increases. Limiting the file size means adding two new rules to those used in TRIBES: a rule to choose which non-dominated solutions should remain in the file and another to adjust the file size in an adaptive manner. The swarm reset process occurs if, after two structural swarm readaptations, at least one non-dominated solution is not included in the file. This allows for possible new regions of the search space to be explored, favoring diversity. The file size is defined a function of the number of optimized objectives. In addition, a mechanism that checks if the process has already been reinitialized is added. If so, the file size is adaptively adjusted according to the number

of non-dominated solutions obtained in the last file size modification. In addition to the dominance criterion, the *crowding distance* is also used to choose the particles that should remain in the file [44]. This is expected to guarantee a good distribution of non-dominated solutions along the Pareto front obtained in each iteration of the algorithm. In the case of MO-TRIBES, the crowding distance is only applied to the best particle of each tribe.

The crowding distance (CD) estimates the size of the largest hypercube around each non-dominated particle i in the file. Thus, the hypercube volume must not contain any other particle than i. Therefore, the objective is to maximize the particles' crowding distance, in order to obtain a Pareto front with the most uniform particle distribution possible. The method based on the crowding distance, as implemented in MO-TRIBES, only allows applications involving up to three objectives [21,45].

In the multi-objective implementation of TRIBES-D (MO-TRIBES), the concept informant particle differs from the concept considered for solving single-objective problems [21]. Given a particle i of position x_i, the first informant is p_i, and is called the cognitive informant. It represents the influence of the particle's own experience on its behavior. At a given instant t, p_i is updated if $x_i(t)$ dominates p_i. Therefore, the dominance criterion is used to evaluate the quality of the particle new position, and hence, to update its cognitive informant. The second informant of particle i is g_i, and is called the social informant. Thus, g_i represents the influence of social experience on the behavior of particle i. In the case of single-objective optimizations, the social informant g is the *shaman* of the tribe to which particle i relates. However, in the case of multi-objective optimizations, the concept of *shaman* differs, as the notion of the best position ever found by the tribe is undefined. Thus, the *shaman* of a tribe is a randomly chosen particle in the tribe at each iteration. The purpose of the *shaman* particle is to act as a leader of the tribe through a communication process with the file's particles. In practice, the choice of g_i differs depending on whether particle i at position x_i is the *shaman*. If this particle is not the *shaman*, g_i is considered the best position reached by the *shaman*; otherwise, g_i is randomly chosen from the file, giving preference to non-dominated particles, which allows for more diversity.

6.5. Selection of the Best Solution

The usage of multi-objective algorithms is expected to return a set of solutions that establish a balanced trade-off between the considered conflicting objectives. This actually is the Pareto front reached by the optimization process. Thus, one must select a single solution to be finally used to optimize the application at hand. In order to choose the best solution for the proposed multi-objective optimization problem, of course the selected solution must belong to the set of Pareto-optimal solutions obtained. Possible selection criteria give preference to a point on the Pareto front that has one of the following characteristics [46]:

- Cr_1: lowest mean-square of normalized objectives' values;
- Cr_2: smallest arithmetic average of the normalized objectives' values;
- Cr_3: smallest geometric mean of the normalized objectives' values.

Before selection of the best solution occurs, data normalization is necessary, since the objectives defined for the problem have values with very different magnitudes to avoid giving preference to the objective whose values have greater magnitude. In this case, the global power consumption of the cooling system is in the order of hundreds of thousands of Watts and is multi-dimensional depending on the refrigeration configuration. On the other hand, the effectiveness of the cooling tower is one-dimensional and varies from 0 to 1. Thus, the objective values must be normalized from 0 to 1. For this purpose, we normalize the system's effectiveness metric and the overall power consumption. In the normalization process, the maximum and the minimum of the range are with respect to the solutions in the Pareto front, which is the final contents of the archive as yielded by the optimization process.

In the energy efficiency MOP, as modeled in this work, the system's effectiveness is maximized while the system's overall power consumption is minimized. Thus, the three criteria defined for selecting the best solution can be defined as in Equations (13) and (14):

$$S^*_{Cr_1} = \min_F \left(\sqrt{\frac{w_1}{\epsilon^2_{a_n}} + w_2 P^2_{global_n}} \right), \tag{13}$$

$$S^*_{Cr_2} = \min_F \left(\frac{w_1/\epsilon_{a_n} + w_2 P_{global_n}}{2} \right) \quad \text{and} \quad S^*_{Cr_3} = \min_F \left(\sqrt{\frac{w_1}{\epsilon_{a_n}} \cdot w_2 P_{global_n}} \right), \tag{14}$$

wherein S^* stands for the solution selected from F, which represents all the solutions of the Pareto front, and w_1 and w_2 represent the coefficients used to weigh the problem objectives: the system's effectiveness ϵ_a and the system's power consumption P_{global}. Note that the greater the weight associated with an objective, the greater its importance for the final solution of the problem. In this work, we consider that the two defined objectives are equally important for the efficiency of the refrigeration system, so we set $w_1 = w_2 = 0.5$. Note that there are other selection strategies, such as those using the ideal and anti-ideal concepts [47] in multi-criterion decision-making.

The strategy of adopting equal weights for the objectives, i.e., considering the conflicting objectives equally important, is coherent. This is based on the fact that a cooling tower operating with low effectiveness consumes a greater amount of water, because the thermal exchange is not carried out properly, which causes water losses by dragging or dripping when filling the tower. The tower's operation with low effectiveness means that the electrical energy consumed by the fans and pumps of the condensing water circuit is not being spent efficiently, since the heat rejected by the tower is below what could be realized. Furthermore, in the current conjuncture where energy resources are increasingly scarce, water is essential to obtain energy, and its waste must be avoided both for economic and environmental reasons. In addition, the increase in water loss in a process based on cooling towers causes an increase in humidity in the environment, which is harmful to the operation of other equipment located in its vicinity. In order to be able to determine the criterion to use, we analyze the results returned by the three criteria. Figure 5 illustrates the application of each of the criteria.

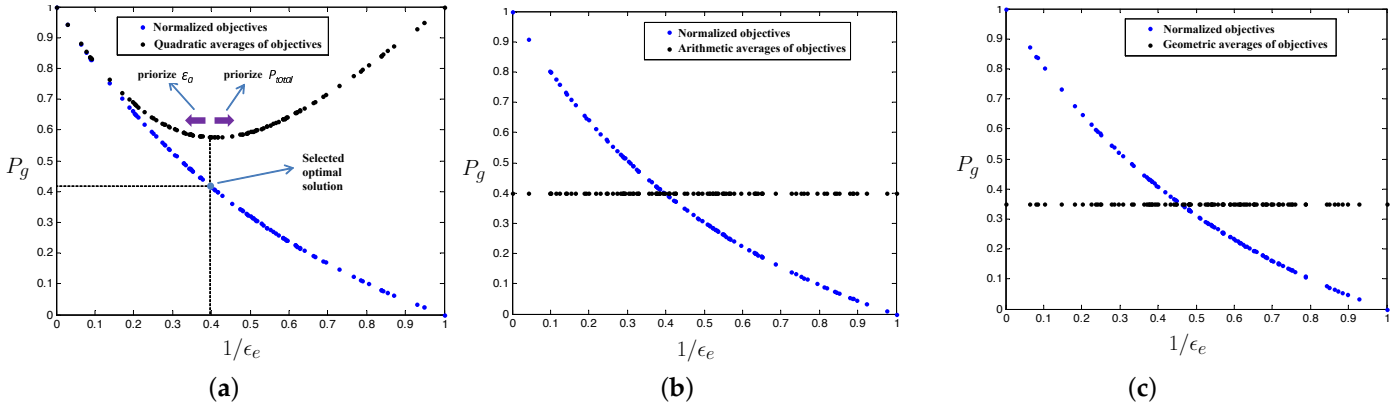

Figure 5. Illustration to motivate the use of the three criteria for selection of the best solution: (**a**) Cr_1; (**b**) Cr_2; (**c**) Cr_3.

In this work, the lowest mean-square (Cr_1) is used as a decision criterion for choosing the optimal solution. In Figure 5a, the minimum point of the mean-square curve of the normalized objectives can be used as a separator between the regions that favor one objective over the other. As one moves towards the left of the minimum point of this curve, one has an increasing preference for the objective of maximizing ϵ_a, which is obtained by means of accentuating the increase in P_{global}. On the other hand, as one moves to the

right of this point, there is an increasing preference for the objective of minimizing P_{global}, causing an augmenting increase in $1/\epsilon_a$, since the reduction in the power of the chillers and tower fans tends to cause an increase in the temperature of the condensing water. On the other hand, considering Figure 5a,b, it can be observed that the application of both methods does not allow identifying a point on the Pareto front that establishes a neat separation between the regions that favor one objective over the other, unlike what occurs in Figure 5a. Based on this analysis, the criterion to be used in this work to choose the optimal solution of the proposed multi-objective problem will be the lowest mean-square of the normalized objectives. Note that the results used in Figure 5 are those obtained by MOPSO for one the operational points.

7. Performance Results

This section presents the results obtained when applying the MOPSO and MO-TRIBES algorithms to solve the proposed multi-objective problem regarding the refrigeration system. The results regarding the two investigated models adopted for the compression chiller are presented and discussed. We also report and analyze the performance of the optimization process with two stopping criteria: either after 50 iterations or after 90 s.

The first stopping criterion was determined experimentally, during the algorithm testing stage. We verify that this number of iterations is sufficient to obtain a Pareto front with a good distribution and enough points to choose the best solution to be applied onto the refrigeration system's cooling towers, fans, and chillers. The second stopping criterion is defined based on the transport delay of the system being optimized. The transport delay is the time interval required to achieve system stability after defining a new setpoint. In this case, we found out that after adjusting a new speed setpoint for the tower fans, on average, 15 min is needed to stabilize a new temperature value for the condensation water. As the thermal system to be optimized has a high transport delay, it is important that the optimization system takes the shortest possible time interval to obtain the optimal solution. Whenever this time interval approaches the transport delay, the chosen solution to be applied may no longer correspond to the best alternative. For instance, assuming that the optimization system obtains the optimal solution in a time interval equal to the transport delay, only after 30 min would we be able to configure a new setpoint for the fan speed. Moreover, due to a possible variation of the thermal load after this time interval, the speed setpoint obtained would possibly no longer correspond to the optimal solution at that instant. Thus, we consider that the decision support system must obtain the optimal solution for the system in a maximum time interval equivalent to 10% of the transport delay, i.e., after 90 s.

During optimization, the algorithms were applied to a real system database. The data were collected in the field using the existing supervisory control and data acquisition (SCADA). A total of 21,385 operational points were collected: 1 point every 5 s. The overall time dedicated to data collection was 29 h and 42 min of operation. The data were collected on different days and times, in order to contemplate different conditions of thermal load and different weather conditions. The values of the wet bulb temperature at the site were collected from the database of The *Instituto Nacional de Pesquisas Espaciais* (INPE), available at [48]. The data were recorded by the meteorological station at Santos Dumont airport in Rio de Janeiro/Brazil, which is located 1.48 km from the site where the cooling towers are installed.

The data provided by INPE present the wet bulb temperatures at the location, T_{BU}, measured at each hour interval. Thus, as the modeling adopted for the cooling tower depends on T_{BU}, it follows that, inevitably, the optimization algorithms had to be applied at every one hour interval using the collected database. Note that this same strategy was adopted in [9,15–17,49]. The use of an interval of one hour between the implementations of the optimizations is justified based on the fact that the variation of thermal load, in buildings such the one explored in this work, only is expressive after an interval of 1 h.

Due to the fact that field data are collected on different days and times, it is possible to use 35 operational points to implement the optimizations, even though the total time interval of the collected database corresponds to 29 h and 42 min. This is so because this duration corresponds to the total of times in which the field data correspond to the timing of the records made available by INPE for the local weather conditions, since the implementation of the optimizations depends on the wet bulb temperature. Thus, using the collected data, in order to perform the optimization at every one hour interval, 35 optimizations were performed with the operational collected data about the 35 operational points. The data of these points are available in Appendix A of [2].

In order to illustrate and analyze the results and Pareto front obtained by the optimization algorithms, we used points 8, 16, and 26 among the 35 operational points indicated in [2]. The choice of these points was based on the fact that they represent very different load situations compared to the other points. This allowed us to verify the effectiveness of the algorithms in diverse situations. Table 3 presents the data for the illustrative operational points. Note that the temperatures are given in °C and the flows in kg/s. Additionally, observe that in the period in which the field data are collected, a maximum of two chillers were used, since the other chillers operate only in the hottest summers, remaining available in the other periods as reserves or redundancies. This situation had already been foreseen prior to data collection. It does not affect the evaluation of the results, since the database used in the optimizations for the 21,385 points corresponds to the same database used to validate the models of the equipment. Note that for statistical significance, we applied the χ-squared test and verified that the null hypothesis is rejected with 0.05 level for all the results.

Table 3. Collected data for the representative operational points used in the discussion.

Point n°	#Chillers	$\dot{m}_{water_{in}}$	$T_{water_{in}}$	T_{BU}	$\dot{m}_{water_{ev}}$	$T_{ae_{ev}}$	$T_{as_{ev}}$
8	2	87.02	29.68	22.94	130.53	10.41	6.01
16	2	70.71	29.40	24.49	106.07	11.16	6.27
26	1	51.65	27.94	23.31	154.96	9.72	6.05

7.1. Results Based on Z_{CAP_1}

Recall that for the Z_{CAP_1} modeling for the chillers, a single search variable is considered: the optimal speed setpoint of the cooling tower fans. The optimal speed obtained is then applied to all fans that are in operation. This procedure is carried out in order to avoid possible resonance scenarios with the tower structure.

7.1.1. MOPSO

The results were obtained using the MATLAB implementation of the MOPSO algorithm (the code of MOPSO algorithm is available in [50], which is based on the MOPSO version presented in [35]). The following parameter settings were used: swarm size 100, external file size 100, cognitive and social coefficients $c_1 = c_2 = 2.05$, and number of grids 10. The values of parameters c_1 and c_2 were defined as suggested by [35]. The remaining parameters were set after experiments in order to obtain a Pareto front with good distribution for the shortest possible convergence time for the algorithm. It is noteworthy to point out that for values of swarm size, external file size, and grid number higher than indicated, maintaining the same number of iterations allowed only for an increase in the algorithm's convergence time, but no improvements in the Pareto front or variations in the results.

Table 4 presents the optimal solutions and respective results for the objective functions obtained for the three illustrative operational points. The results for all 35 operational points used in the optimization implementations are available in Appendix A of [2]. Recall that n_{best} represents the optimal solution obtained by the optimization algorithm, given in Hz, ϵ_a represents the effectiveness obtained for the cooling tower, P_{global} represents the global power demanded by fans and chillers of the cooling system, in kW, and *ec* (%) represents the percentage savings in global energy consumption.

Table 4. Optimal solutions obtained by MOPSO for the 3 operational points using Z_{CAP_1}.

Point n°	MOPSO—50 Iterations				MOPSO—90 s			
	n_{best}	ϵ_a	P_{global}	ec (%)	n_{best}	ϵ_a	P_{global}	ec (%)
8	46.52	0.5231	835.90	4.04	45.71	0.5217	834.17	4.24
16	47.71	0.6941	963.68	3.54	47.09	0.6913	962.34	3.67
26	46.97	0.6854	365.60	10.28	48.03	0.6904	367.07	9.92

Table 5 presents the optimal solutions obtained that guarantee the established restrictions for the three illustrative operational points. The results for all 35 operational points used in the optimization implementations are available in Appendix B of [2]. Recall that $T_{ae_{co}}$ represents the predicted temperature for the water of the condensation circuit that leaves the cooling tower and goes towards the chillers, and $T_{as_{co}}$ represents the predicted temperature for the water in the condenser circuit that leaves the chillers and travels towards the cooling tower.

Table 5. Verification of compliance of MOPSO with the operating restrictions of the equipment for the 3 operational points when using Z_{CAP_1} modeling.

Point n°	MOPSO—50 Iterations				MOPSO—90 s			
	n_{best}	$T_{ae_{co}}$	ΔT	$T_{as_{co}}$	n_{best}	$T_{ae_{co}}$	ΔT	$T_{as_{co}}$
8	46.52	26.15	3.21	30.74	45.71	26.16	3.22	30.75
16	47.71	25.99	1.50	32.52	47.09	26.01	1.52	32.53
26	46.97	24.77	1.46	29.15	48.03	24.74	1.43	29.12

Figure 6 shows the Pareto fronts obtained after 50 iterations, and Figure 7 shows the Pareto fronts obtained after 90 s for the three operational points. The circled points represent the selected optimal solutions.

In Table 4, comparing the results obtained with the stopping criterion of 90 s to those obtained after 50 iterations, it can be verified that operational point 8 obtained an increase of 0.2% in the economy of the system's overall power requirements at the expense of a 0.14% reduction in tower effectiveness. For operational point 16, there is an increase of 0.13% in global power savings for a 0.28% reduction in effectiveness. For point 26, there is a reduction of 0.36% in the overall power savings for an increase of 0.5% in the tower's effectiveness.

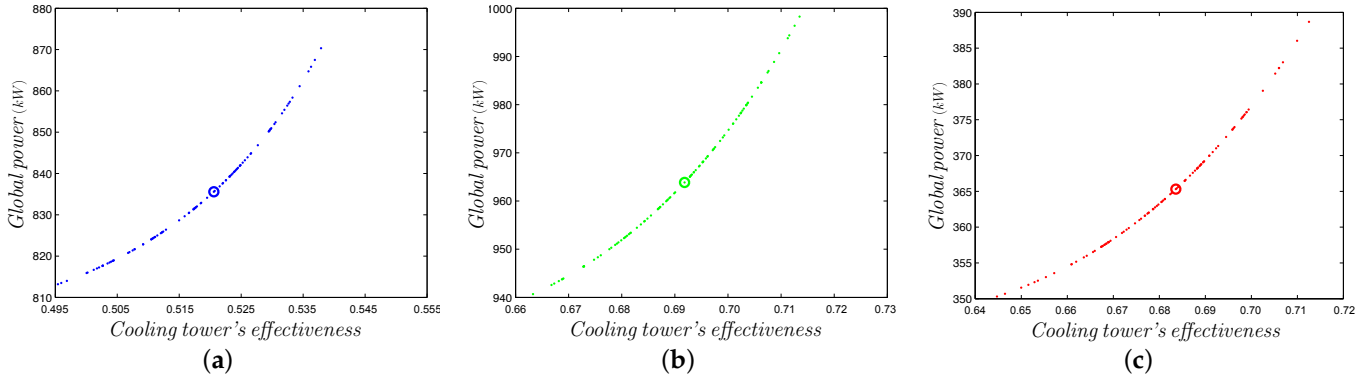

Figure 6. Pareto fronts $P_{global} \times \epsilon_a$ and optimal solutions obtained by MOPSO for the 3 operational points using the Z_{CAP_1} with the stopping criterion of 50 iterations. (**a**) Point n°8. (**b**) Point n°16. (**c**) Point n°26.

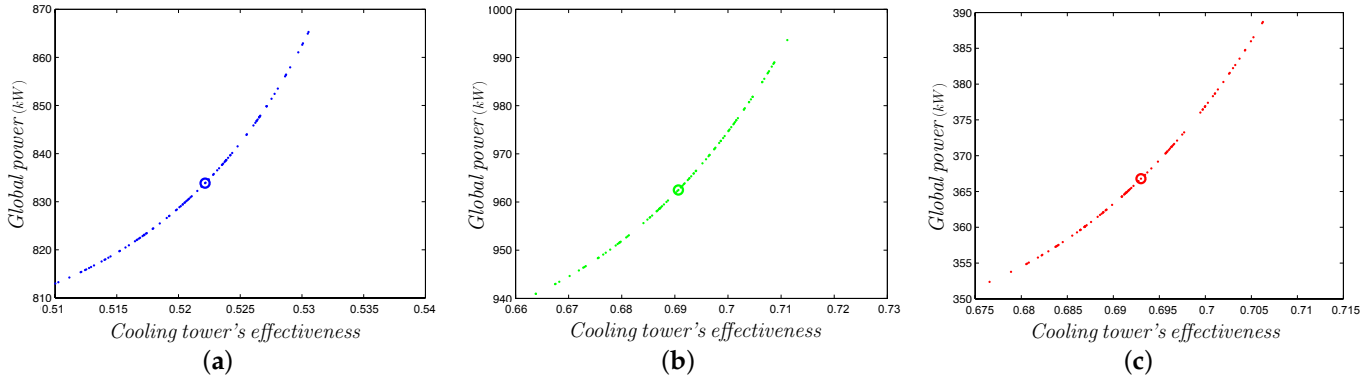

Figure 7. Pareto fronts $P_{global} \times \epsilon_a$ and optimal solutions obtained by MOPSO for the 3 operational points using the Z_{CAP_1} with the stopping criterion of 90 s. (**a**) Point $n°8$. (**b**) Point $n°16$. (**c**) Point $n°26$.

Figure 6 shows the Pareto front for the 50-iterations stopping criterion, and Figure 7 presents that regarding the 90 s stopping criterion. Note that the obtained Pareto fronts are different, and the displacement of the optimal solution obtained after 50 iterations can be clearly seen in a different point in the optimal solution obtained after 90 s, either favoring energy savings, as for points 8 and 16, or favoring the effectiveness of the tower, as in the case of point 26. After 90 s, only point 16 converged to a solution in which there was an increase in energy savings lower than the reduction in the effectiveness of the tower, which is justified by the fact that the different stopping criteria are different applications of the algorithm. The optimization process stopping after 90 s allowed continuation after 50 iterations. In fact, the convergence of the algorithm only occurred with the stopping criterion of 90 s; note that the 50 iterations were not enough for the actual convergence of the algorithm.

7.1.2. MO-TRIBES

For MO-TRIBES, the results were obtained using a C++ implementation available in [51]. The following parameters were used: maximum number of particles per tribe was set to 10, and maximum number of tribes to 40. The adaptation frequency was set to 5 iterations. The parameter values were obtained based on experiments, in order to obtain a Pareto front with good distribution and the lowest possible convergence time. It is noted that there were no other convergence parameters, as in the case of MOPSO. The only requirement consisted of defining the swarm size. This is one advantage of MO-TRIBES when compared to MOPSO.

Table 6 presents the optimal solutions and respective results for the objective functions obtained for the three illustrative operational points. The results for all 35 operational points used in the optimization implementations are available in Appendix A of [2]. As before, n_{best} represents the optimal solution obtained by the MOPSO algorithm, given in Hz, ϵ_a represents the effectiveness obtained for the cooling tower, P_{global} represents the global power demanded by fans and chillers of the cooling system, in kW, and ec (%) represents the percentage savings in global energy consumption.

Table 6. Optimal solutions obtained by MO-TRIBES for the 3 operational points using Z_{CAP_1}.

Point $n°$	MO-TRIBES—50 Iterations				MO-TRIBES—90 s			
	n_{best}	ϵ_a	P_{global}	ec (%)	n_{best}	ϵ_a	P_{global}	ec (%)
8	45.90	0.5220	834.58	4.19	45.99	0.5222	834.77	4.17
16	46.37	0.6880	960.84	3.82	46.37	0.6880	960.84	3.82
26	46.53	0.6833	365.01	10.42	46.56	0.6834	365.05	10.41

Table 7 presents the optimal solutions obtained that guarantee the established restrictions for the three illustrative operational points. The results for all 35 operational points used in the optimization implementations are available in Appendix B of [2]. As before, $T_{ae_{co}}$ represents the predicted temperature for the water of the condensation circuit that leaves the cooling tower and goes towards the chillers, and $T_{as_{co}}$ represents the predicted temperature for the water in the condenser circuit that leaves the chillers and travels towards the cooling tower.

Table 7. Verification of compliance of MO-TRIBES with the operational restrictions of the equipment for the 3 operational points when using Z_{CAP_1} modeling.

Point n°	MO-TRIBES—50 Iterations				MO-TRIBES—90 s			
	n_{best}	$T_{ae_{co}}$	ΔT	$T_{as_{co}}$	n_{best}	$T_{ae_{co}}$	ΔT	$T_{as_{co}}$
8	45.90	26.16	3.22	30.74	45.99	26.16	3.22	30.74
16	46.37	26.02	1.53	32.55	46.37	26.02	1.53	32.55
26	46.53	24.78	1.47	29.16	46.56	24.78	1.47	29.16

Figure 8 shows the Pareto fronts obtained after 50 iterations, and Figure 9 shows the Pareto fronts obtained after 90 s for the three operational points. The circled points represent the selected optimal solutions.

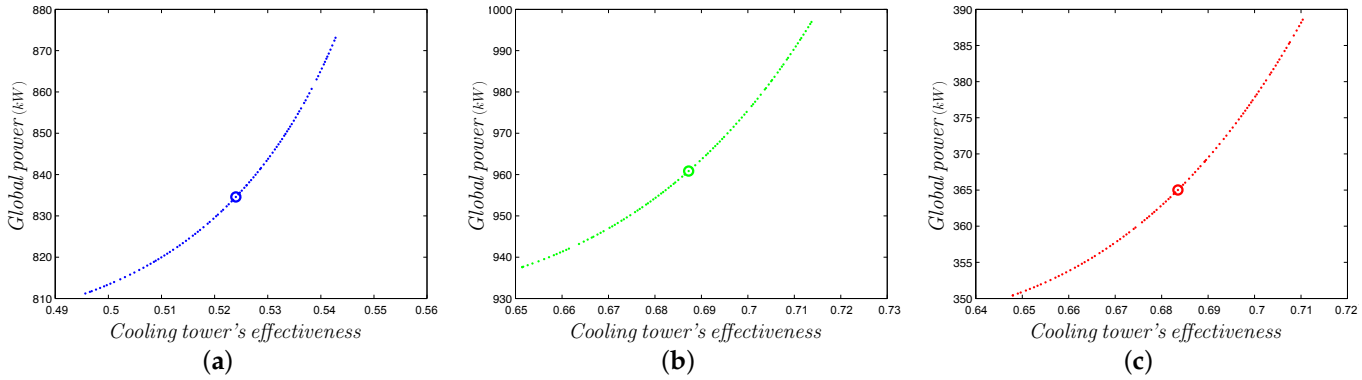

Figure 8. Pareto fronts $P_{global} \times \epsilon_a$ and optimal solutions obtained by MO-TRIBES for the 3 operational points using the Z_{CAP_1} with the stopping criterion of 50 iterations. (**a**) Point n°8. (**b**) Point n°16. (**c**) Point n°26.

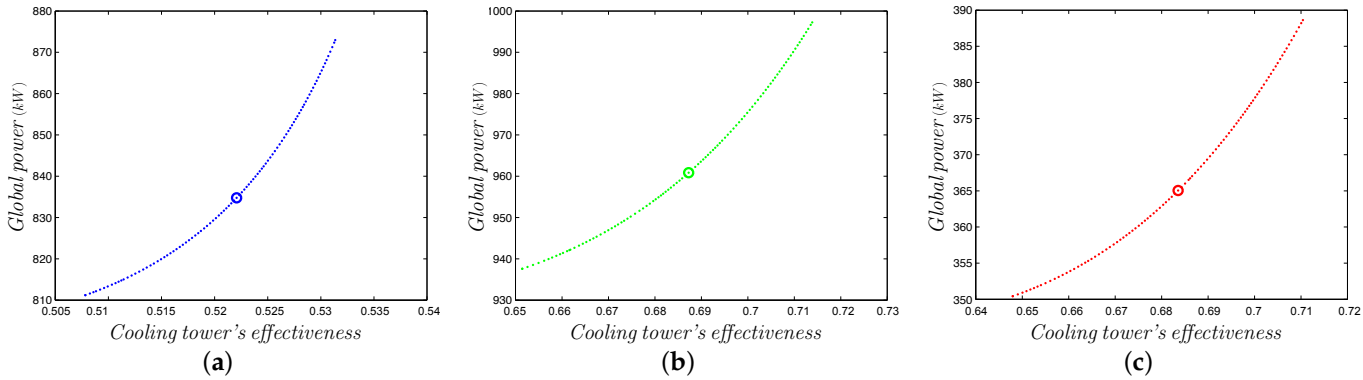

Figure 9. Pareto fronts $P_{global} \times \epsilon_a$ and optimal solutions obtained by MO-TRIBES for the 3 operational points using the Z_{CAP_1} with the stopping criterion of 90 s. (**a**) Point n°8. (**b**) Point n°16. (**c**) Point n°26.

In Table 6, when comparing the results obtained for the two stopping conditions, good proportionality between the increases and reductions of effectiveness and electric

energy savings can be observed. Regarding the results obtained with 50 iterations, after 90 s, operational point 8 exhibited a reduction of 0.02% in the global power savings for an increase of exactly 0.02% in the tower's effectiveness. Similarly, operational point 26 shows a reduction of 0.01% in power consumption for an increase of exactly 0.01% in effectiveness. Finally, operational point 16 did not present any variation in the result for a greater number of iterations, verifying the full convergence of the algorithm.

In Figures 8 and 9, it can be noted that the obtained Pareto fronts for both stopping criteria are practically identical. We verified excellent distribution and continuity of the reached Pareto fronts compared to those achieved by MOPSO. In addition, since the optimal solutions converged to the same region for each of the considered operational points, and given the minimal variations presented by the obtained results, the proper convergence of the algorithm was verified using the two established stopping criteria.

7.2. Results Based on Z_{CAP_2}

In the Z_{CAP_2} modeling in which the Z_{CAP_2} factor was to estimate the power demanded by the chillers, two search variables were considered: the optimal setpoint for the speed of the fans of the cooling towers and the optimal setpoint for the temperature of the evaporator outlet water. The results were obtained from the application of the optimal solution found by each algorithm, in order to allow comparison between the results and the Pareto fronts obtained for the two established stopping criteria. In this case, for the analysis of the results, the same operational points, i.e., 8, 16, and 26, were used for the evaluation of the Z_{CAP_2} modeling.

7.2.1. MOPSO

As stated before, the results were obtained using the MATLAB implementation of the MOPSO algorithm available in [35,50]. The same parameter settings defined in Section 7.1.1 were used. Note that the adoption of the same parameter settings allowed the comparison of the results obtained by the optimizations implemented using the two models: Z_{CAP_1} and Z_{CAP_2}.

Table 8 presents the optimal solutions and respective results for the objective functions obtained for the three illustrative operational points. The results for all 35 operational points used in the optimization implementations are available in Appendix A of [2]. Recall that n_{best} represents the optimal solution obtained by the optimization algorithm, given in Hz, T_{best} represents the optimal temperature for the chilled water that leaves the evaporators of the chillers, given in °C, ϵ_a represents the effectiveness obtained for the cooling tower, P_{global} represents the global power demanded by the fans and chillers of the cooling system, in kW, and *ec* (%) represents the percentage savings in global energy consumption.

Table 8. Optimal solutions obtained by MOPSO for the 3 operational points using Z_{CAP_2}.

Point *n*°	MOPSO—50 Iterations					MOPSO—90 s				
	n_{best}	T_{best}	ϵ_a	P_{global}	*ec* (%)	n_{best}	T_{best}	ϵ_a	P_{global}	*ec* (%)
8	43.16	6.45	0.5173	830.57	12.64	45.36	6.45	0.5211	835.10	12.12
16	60.00	6.96	0.7452	927.78	12.04	60.00	6.85	0.7452	944.41	10.37
26	60.00	6.65	0.7434	382.53	13.31	57.05	6.84	0.7309	367.27	17.05

Table 9 presents the optimal solutions obtained that guarantee the established restrictions for the three illustrative operational points. The results for all 35 operational points used in the optimization implementations are available in Appendix B of [2]. As before, $T_{ae_{co}}$ represents the predicted temperature for the water of the condensation circuit that leaves the cooling tower and goes towards the chillers, and $T_{as_{co}}$ represents the predicted temperature for the water in the condenser circuit that leaves the chillers and travels towards the cooling tower.

Table 9. Verification of compliance of MOPSO with the operational restrictions of the equipment for the 3 operational points when using Z_{CAP_2} modeling.

Point $n°$	MOPSO—50 Iterations					MOPSO—90 s				
	n_{best}	T_{best}	$T_{ae_{co}}$	ΔT	$T_{as_{co}}$	n_{best}	T_{best}	$T_{ae_{co}}$	ΔT	$T_{as_{co}}$
8	43.16	6.45	26.19	3.25	30.79	45.36	6.45	26.17	3.23	30.76
16	60.00	6.96	25.74	1.25	31.75	60.00	6.85	25.74	1.25	31.87
26	60.00	6.65	24.50	1.19	28.77	57.05	6.84	24.56	1.25	28.72

Figure 10 shows the Pareto fronts obtained after 50 iterations, and Figure 11 shows the Pareto fronts obtained after 90 s for the three operational points. The circled points represent the selected optimal solutions.

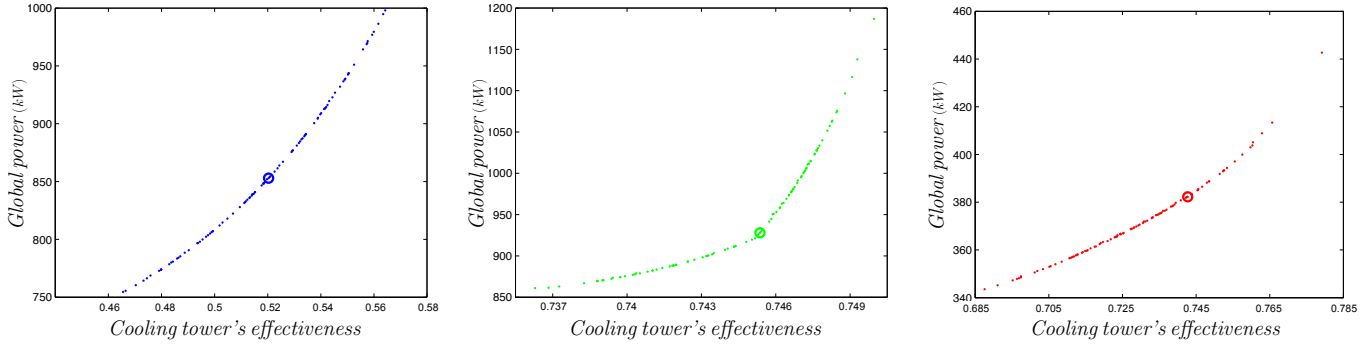

Figure 10. Pareto fronts $P_{global} \times \epsilon_a$ and optimal solutions obtained by MO-TRIBES for the 3 operational points using the Z_{CAP_1} with the stopping criterion of 50 iterations. (**a**) Point $n°8$. (**b**) Point $n°16$. (**c**) Point $n°26$.

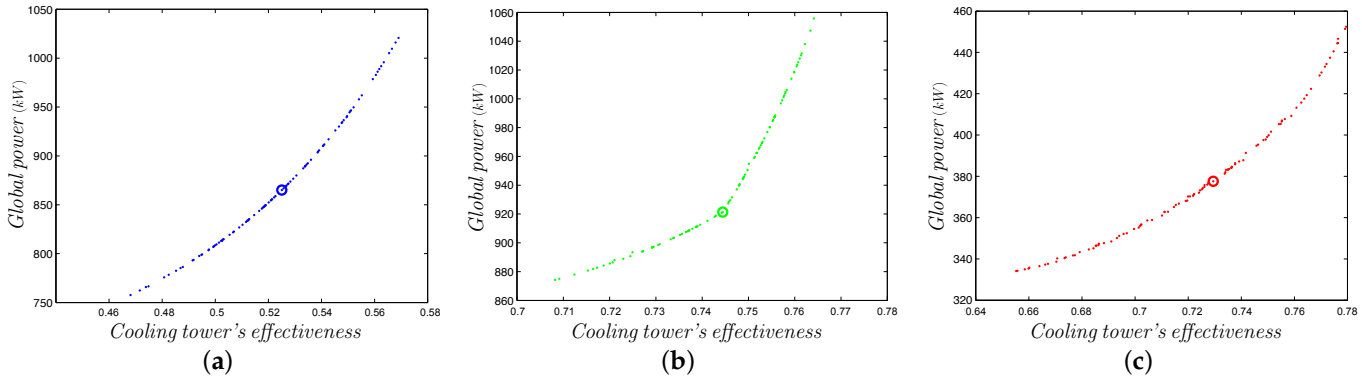

Figure 11. Pareto fronts $P_{global} \times \epsilon_a$ and optimal solutions obtained by MO-TRIBES for the 3 operational points using the Z_{CAP_1} with the stopping criterion of 90 s. (**a**) Point $n°8$. (**b**) Point $n°16$. (**c**) Point $n°26$.

When we compare the results obtained with the stopping criterion of 90 s with those obtained after 50 iterations in Table 8, we can note that operational point 8 presented a reduction of 1.44% in the economy of the system's overall power at the expense of an increase of only 0.22% in the tower's effectiveness. Operational point 16 presented an increase of 0.66% in the global power savings, while maintaining the same value of effectiveness obtained with 50 iterations. For operational point 26, there was a 1.31% increase in global power savings, at the expense of a 3.19% reduction in the tower's effectiveness. Moreover, by comparing the Pareto fronts in Figures 10 and 11, we can observe those obtained after 90 s differ little from those obtained after 50 iterations. A clear shift from the optimal solution obtained after 50 iterations to a new position after using

the stopping criterion of 90 s can also be noted. In this case, as indicated, there are points where the savings increase is lower than the tower's effectiveness reduction.

7.2.2. MO-TRIBES

Recall that for MO-TRIBES, the results were obtained using a C++ implementation available in [51]. The same parameter settings defined in Section 7.1.2 were used. Note that the adoption of the same parameter setting allows a comparison between the results obtained by the optimizations implemented using the two models: Z_{CAP_1} and Z_{CAP_2}.

Table 10 presents the optimal solutions and respective results for the objective functions obtained for the three illustrative operational points. The results for all 35 operational points used in the optimization implementations are available in Appendix A of [2]. As before, n_{best} represents the optimal solution obtained by the MO-TRIBES algorithm, given in Hz, T_{best} represents the optimal temperature for the chilled water that leaves the evaporator of the chillers, given in °C, ϵ_a represents the effectiveness obtained for the cooling tower, P_{global} represents the global power demanded by fans and chillers of the cooling system, in kW, and ec (%) represents the percentage savings in global energy consumption.

Table 10. Optimal solutions obtained by MO-TRIBES for the 3 operational points using Z_{CAP_2}.

Point n°	MO-TRIBES—50 Iterations					MO-TRIBES—90 s				
	n_{best}	T_{best}	ϵ_a	P_{global}	ec (%)	n_{best}	T_{best}	ϵ_a	P_{global}	ec (%)
8	44.52	6.31	0.5197	852.37	10.14	44.29	6.31	0.5193	851.93	10.19
16	60.00	6.98	0.7452	924.78	12.34	60.00	6.98	0.7452	923.47	12.47
26	57.32	6.77	0.7321	371.08	16.12	58.09	6.83	0.7354	369.68	16.46

Table 11 presents the optimal solutions obtained that guarantee the established restrictions for the three illustrative operational points. The results for all 35 operational points used in the optimization implementations are available in Appendix B of [2]. As before, $T_{ae_{co}}$ represents the predicted temperature for the water of the condensation circuit that leaves the cooling tower and goes towards the chillers, and $T_{as_{co}}$ represents the predicted temperature for the water in the condenser circuit that leaves the chillers and travels towards the cooling tower.

Table 11. Verification of compliance of MO-TRIBES with the operational restrictions of the equipment for the 3 operational points when using Z_{CAP_2} modeling.

Point n°	MO-TRIBES—50 Iterations					MO-TRIBES—90 s				
	n_{best}	T_{best}	$T_{ae_{co}}$	ΔT	$T_{as_{co}}$	n_{best}	T_{best}	$T_{ae_{co}}$	ΔT	$T_{as_{co}}$
8	44.52	6.31	26.18	3.24	30.88	44.29	6.31	26.18	3.24	30.88
16	60.00	6.98	25.74	1.25	31.73	60.00	6.98	25.74	1.25	31.72
26	57.32	6.77	24.55	1.24	28.75	58.09	6.83	24.54	1.23	28.70

Figure 12 shows the Pareto fronts obtained after 50 iterations, and Figure 13 shows the Pareto fronts obtained after 90 s for the three operational points. The circled points represent the selected optimal solutions.

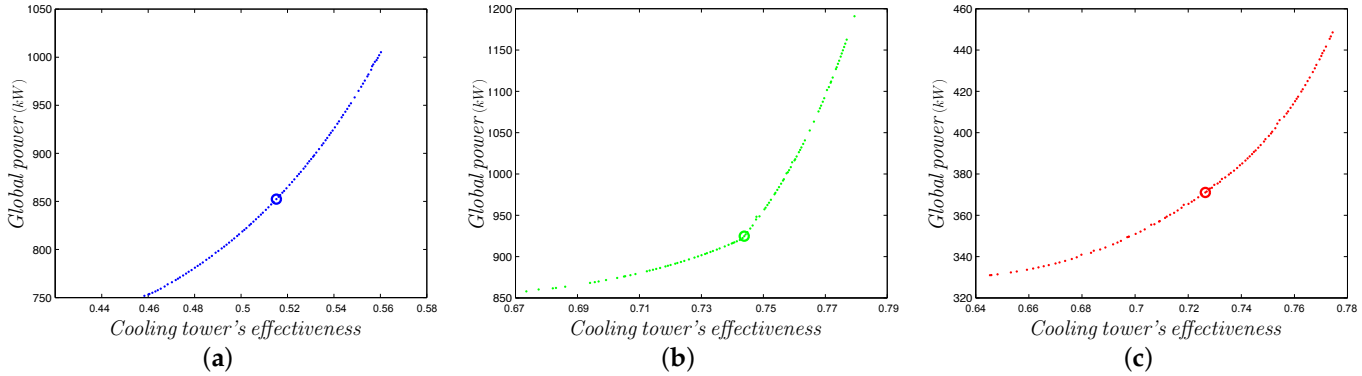

Figure 12. Pareto fronts $P_{global} \times \epsilon_a$ and optimal solutions obtained by MO-TRIBES for the 3 operational points using the Z_{CAP_1} with the stopping criterion of 50 iterations. (**a**) Point $n°8$. (**b**) Point $n°16$. (**c**) Point $n°26$.

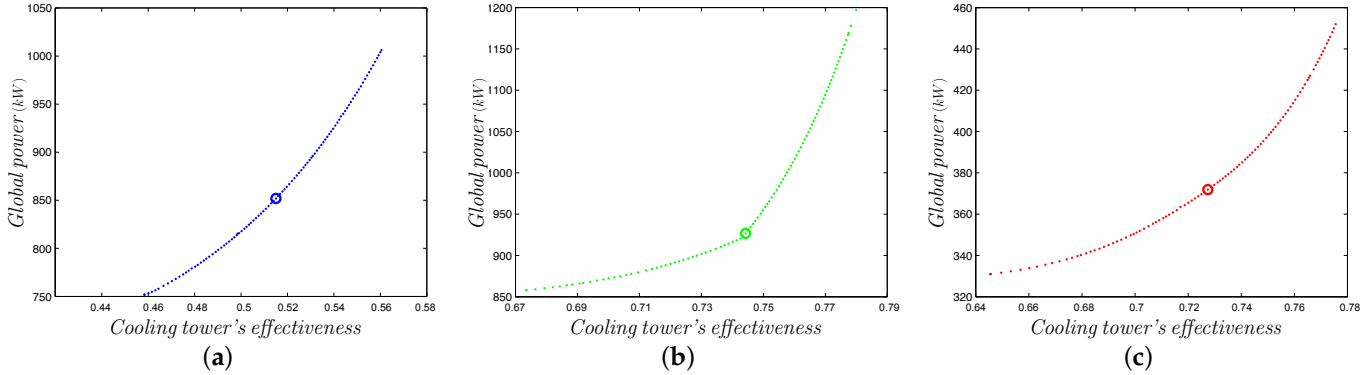

Figure 13. Pareto fronts $P_{global} \times \epsilon_a$ and optimal solutions obtained by MO-TRIBES for the 3 operational points using the Z_{CAP_1} with the stopping criterion of 90 s. (**a**) Point $n°8$. (**b**) Point $n°16$. (**c**) Point $n°26$.

When we compare the results obtained with the stopping criterion of 90 s with those obtained after 50 iterations in Table 10, we have that operational point 8 allowed for a reduction of 0.04% in the tower's effectiveness, obtaining a 0.05% increase in the overall power savings of the system. Point 16 presents an increase in 0.13% in the global power savings, maintaining the same value of effectiveness obtained after 50 iterations. On the other hand, point 26 yielded an increases in global power savings and in effectiveness of 0.34% and 0.33%, respectively. When comparing the Pareto fronts of Figures 12 and 13, we can note the similarity for the two stopping conditions, confirming the proper convergence of the algorithm.

7.2.3. Performance Evaluation

As a complement to the analysis performed in Sections 7.1 and 7.2, in this section, we compare the performances of the algorithms based on the metric of coverage [52]. It evaluates how much one set of solutions dominates another. Given two Pareto fronts A and B, the coverage $C(A, B)$ indicates the percentage of solutions in B that are weakly dominated by those in A. Coverage is calculated according to Equation (15):

$$C(A, B) = \frac{|b \in B : \exists a \in A, a \succeq b|}{|B|}. \quad (15)$$

The coverage metric provides values in $[0, 1]$. If $C(A, B) = 1$, then all solutions in B are dominated by solutions in A. On the other hand, if $C(A, B) = 0$, then none of the solutions

in B is dominated by any solution in A. To apply this metric, both $C(A, B)$ and $C(B, A)$ must be evaluated, as these values are not complementary.

Table 12 presents the coverage values of the obtained Pareto front when using the Z_{CAP_1} modeling. We observe that the values obtained by comparing all the applied algorithms are always very close to zero, for both considered stopping criteria. This result was expected, since the search space is reduced to a single dimension, represented by the speed of the cooling tower fans. This aids the convergence of the algorithms. Thus, most obtained Pareto fronts overlap. Note that the values presented in Table 12 correspond to the average of the coverage value for the fronts obtained for the 35 operational points indicated in Appendix A of [2].

Table 12. Average values for the coverage for the 35 operational points, regarding both models and both stopping criteria.

Model	Stopping Criterion	Algorithm	MOPSO	MO-TRIBES
Z_{CAP_1}	50 iterations	MOPSO	–	0.0142
		MO-TRIBES	0.0845	–
	90 s	MOPSO	–	0.0217
		MO-TRIBES	0.0769	–
Z_{CAP_2}	50 iterations	MOPSO	–	0.0415
		MO-TRIBES	0.0462	–
	90 s	MOPSO	–	0.0269
		MO-TRIBES	0.0462	–

Nevertheless with a close look into the values of Table 12, we can note that MO-TRIBES presents indices a little higher than MOPSO. However, as the values obtained for this metric are very low for both algorithms, it follows that only after applying each chosen optimal solution to the dataset of the system to be optimized, one can in fact identify the overall best solution. This comparison would be based on the average values obtained for energy savings and reduction in tower effectiveness. Based on the coverage values achieved, we can safely conclude that the algorithms achieve similar energy efficiency.

8. Performance Comparison

In this section, the results obtained after the application of the MOPSO and MO-TRIBES multi-objective optimization algorithms are compared. Furthermore, we extend the comparison to there other multi-objective evolutionary algorithms, SPEA2, NSGA-II. and Micro-GA [2,3]. This allows us to further assess the performances of the applied algorithms. Initially, the obtained results are compared with field data to evaluate the gains obtained in terms of energy savings and cooling tower effectiveness. Then, the results obtained by the algorithms are compared with each other in order to choose the best modeling and the best algorithm for the application. The following metrics were used for to back the comparison and selection process:

- Average, minimum, and maximum savings in power consumption of the system;
- Average, minimum, and maximum effectiveness obtained for the cooling tower;
- Ratio between the average savings obtained in the global power consumption of the system and the respective reduction in the average effectiveness verified for the tower;

where the average values indicated were calculated after applying the optimization results to the collected 21,385 operational points. In this work, a third metric is used, called the energy efficiency factor (EEF). It is computed using Equation (16):

$$EEF = \frac{\text{Global power consumption average savings}}{\text{Cooling tower average effectiveness reduction}} \quad (16)$$

8.1. First Model

Considering the Z_{CAP_1} model, Table 13 presents the results of MOPSO and MO-TRIBES, regarding both stopping criteria, and those obtained for NSGA-II, SPEA2, and Micro-GA. The values indicated refer to the application of the results obtained for the 35 operational points, as presented in Appendix A of [2], to the 21,385 actual field data points collected from the real refrigeration system. The execution time (T) indicates the average execution time of the algorithm with the stopping criterion of 50 iterations, and is given in seconds. For the 90-s case, we report the number of required iterations (I) instead, as the execution time is 90 s.

Table 13. Power saving (%), effectiveness, time (s) and number of iterations for the Z_{CAP_1} model.

Metric	After 50 Iterations					After 90 s				
	MOPSO	**MO-TRIBES**	**NSGA-II**	**SPEA2**	**MicroGA**	**MOPSO**	**MO-TRIBES**	**NSGA-II**	**SPEA2**	**MicroGA**
PS_{avg}	4.99	**5.27**	5.20	5.22	5.23	5.01	**5.27**	5.20	5.22	5.24
PS_{min}	−3.39	**−2.76**	−2.80	−2.82	−2.80	**−2.70**	−2.76	−2.76	−2.82	−2.84
PS_{max}	16.02	**16.43**	16.34	16.36	16.32	15.96	**16.43**	16.24	16.37	16.32
ϵ_{avg}	**0.6134**	0.6105	0.6111	0.6110	0.6109	**0.6141**	0.6104	0.6110	0.6110	0.6108
ϵ_{min}	**0.4810**	0.4806	0.4807	0.4808	0.4808	**0.4816**	0.4806	0.4809	0.4808	0.4811
ϵ_{max}	**0.7950**	0.7882	0.7896	0.7885	0.7891	**0.7919**	0.7882	0.7888	0.7899	0.7888
T/I	47.26	9.36	62.72	**6.19**	43.32	97	292	**71**	752	155

It is noteworthy to point out that, as MOPSO, NSGA-II, and Micro-GA were implemented in MATLAB, the underlying optimization processes of 50 iterations required longer execution times when compared to MO-TRIBES and SPEA2, as the latter were implemented in C++. This is due to the manipulation of arrays performed internally by MATLAB, which increases the execution time when compared to an implementation in C++. This kind of result was expected, and for this reason, this criterion was not used to choose the best algorithm. It was only used for the purpose of verifying the compliance with the time required for the convergence of the algorithms, and it was verified that the 90 s criterion meets the time requirement. For both stopping criteria, we can clearly note that the algorithm that reached the best average and maximum values of power saving was MO-TRIBES. As for the minimum savings values obtained with the database used, MO-TRIBES also provided the best result for 50 iterations, though it was surpassed by MOPSO when stopping after 90 s.

For all algorithms, there are records of negative values of energy savings. We verified that at those points whose fan speed in the collected field data was 30 Hz. As the optimization suggested increasing their speed in order to increase the tower's effectiveness, there was an increase in the electrical energy consumption. In addition, the chillers operate, at these points, with a low thermal load, which makes the consumption of fans more significant. Therefore, the increase in power consumption at these points is consistent and matches the expectations for system optimization. As expected, the algorithm that presents the best result in terms of average economy is the same that has the worst result in terms of tower effectiveness, since these objectives are conflicting. In this case, MO-TRIBES presents the worst average, minimum, and maximum effectiveness. The MOPSO algorithm yielded the best effectiveness, i.e., average, minimum, and maximum, regarding both stopping conditions. However, it had the worst global energy savings, i.e., average and maximum, regarding both stopping criteria.

Considering that the estimated effectiveness of the cooling tower design corresponds to 0.7551, it follows that, ideally, this value should be used as a reference for purposes of comparing the results obtained for the exploited algorithms. However, it is known that the achievement of this value of effectiveness depends not only on the setpoints of the cooling tower operation, but also on external factors, such as ambient temperature and wet bulb temperature, which often makes this effectiveness value difficult to reach. Thus, the reference value for evaluating the algorithms must be the average effectiveness obtained by

applying the 21,385 operational points collected for the cooling tower modeling. This value was evaluated to 0.6761. Hence, the best algorithm for the application must be the one that achieves the highest average global power savings with the least possible detriment to the average effectiveness of the cooling tower, the latter being obtained based on the modeling adopted for the cooling tower.

Table 14 shows the values of the used metrics when the model based on Z_{CAP_1} was used. Therein, the average power savings (APS) (%) represents the energy savings obtained after performing the optimization; $\overline{\epsilon_a}$ represents the average effectiveness; average effectiveness reduction (AER) represents the difference between the average effectiveness obtained after applying the algorithms and the average effectiveness verified without optimization, both obtained by applying the cooling tower modeling to the collected field data. The average effectiveness values are indicated in percentages, and EEF is given by the ratio between the average savings (%), as defined earlier in Equation (16).

Table 14. Selection of the best algorithm for the Z_{CAP_1}-based model considering both stopping criteria.

Metric	After 50 Iterations					After 90 s				
	MOPSO	**MO-TRIBES**	**NSGA-II**	**SPEA2**	**MicroGA**	**MOPSO**	**MO-TRIBES**	**NSGA-II**	**SPEA2**	**MicroGA**
APS	4.99	**5.27**	5.20	5.22	5.23	5.01	**5.27**	5.20	5.22	5.24
$\overline{\epsilon_a}$	**61.34**	61.05	61.11	61.10	61.09	**61.41**	61.04	61.10	61.10	61.08
AER	**6.27**	6.56	6.50	6.51	6.52	**6.20**	6.57	6.51	6.51	6.53
EEF	0.7967	**0.8033**	0.7998	0.8017	0.8014	**0.8072**	0.8023	0.7993	0.8020	0.8017

It is noteworthy that the obtained *EEF* rates are less than one for all algorithms. This means a reduction in effectiveness greater than the obtained average power savings. On the other hand, this result does not mean that the multi-objective optimization using the Z_{CAP_1}-based model is unfeasible, since the average effectiveness values obtained for all algorithms are acceptable in practical terms. Figure 14 allows a visual comparison of the improvement yielded in average power savings for both stopping criteria (APS-50i and APS-90s) in relation to the reduction verified in the average effectiveness for both stopping criteria (AER-5i and AER-90s) of the tower for both applied algorithms.

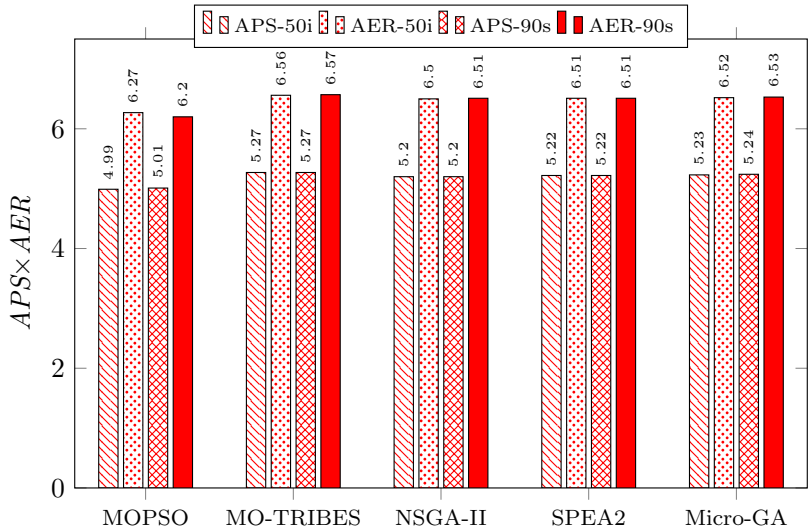

Figure 14. Comparison between average power savings and corresponding average effectiveness reduction regarding the Z_{CAP_1}-based model and both stopping criteria for the applied algorithms.

Based on the established criteria, it appears that MO-TRIBES achieved the best result for the stopping criterion of 50 iterations, followed by SPEA2, MIcro-GA, NSGA-II, and

MOPSO. For the stopping criterion of 90 s, we observe that the MOPSO algorithm outperformed the other four algorithms. This indicates that 50 iterations were not enough to obtain the best result using MOPSO. Thus, for fair comparison purposes, the stopping criterion of 90 s must be considered as a reference to determine the best algorithm for the application in both models Z_{CAP_1} and Z_{CAP_2}. Therefore, using the Z_{CAP_1}-based model, MOPSO is the best algorithm for the application, followed by the MO-TRIBES algorithm. This does not mean that the results yielded by MO-TRIBES, NSGA-II, SPEA2, and Micro-GA are unsatisfactory. Instead, it means that MOPSO is able to reach the best balance between the defined conflicting objectives, presenting optimal solutions that resulted in power savings with the smallest reduction in effectiveness of the cooling towers during the considered period of operation.

8.2. Second Model

We now analyze the performance of the Z_{CAP_2}-based model. This was done using the same methodology as for the Z_{CAP_1}-based model. Table 15 presents the evaluated metrics results of compared algorithms, regarding both stopping criteria. Recall that the values indicated refer to the application of the results obtained for the 35 operational points, as presented in Appendix A of [2], to the 21,385 actual field data points collected from the real refrigeration system. As before, the execution time (T) indicates the average execution time of the algorithm with the stopping criterion of 50 iterations, and is given in seconds, and for the 90 s case, we report the number of required iterations (I) instead, as the execution time is known, i.e., 90 s.

Table 15. Power saving (%), effectiveness, time, and number of iterations for the Z_{CAP_2} model.

Metric	After 50 Iterations					After 90 s				
	MOPSO	**MO-TRIBES**	**NSGA-II**	**SPEA2**	**MicroGA**	**MOPSO**	**MO-TRIBES**	**NSGA-II**	**SPEA2**	**MicroGA**
PS_{avg}	**9.35**	9.19	8.28	8.48	8.43	**9.48**	9.27	8.32	8.50	8.43
PS_{min}	−5.38	−3.17	−4.09	−3.72	−4.64	−7.12	−3.13	−4.26	−3.72	−3.66
PS_{max}	26.75	26.17	25.36	26.07	25.67	26.99	26.21	26.16	25.27	25.46
ϵ_{avg}	0.6222	0.6229	0.6219	0.6232	0.6183	0.6229	0.6233	0.6200	0.6247	0.6159
ϵ_{min}	0.4805	0.4817	0.4826	0.4843	0.4818	0.4797	0.4827	0.4818	0.4833	0.4816
ϵ_{max}	0.8474	0.8474	0.8470	0.8474	0.8350	0.8474	0.8474	0.8466	0.8473	0.8285
T/I	29.64	4.819	69.82	1.570	77.92	151	320	63	548	78

In Table 15, it is noteworthy to confirm that again, MOPSO, NSGA-II, and Micro-GA required longer execution times for 50 iterations than MO-TRIBES and SPEA2, given the fact that the former were implemented in MATLAB and the latter were implemented in C++. For both stopping criteria, it is possible to conclude that the algorithm that achieved the best power savings, i.e., average and maximum, was MOPSO, followed by MO-TRIBES, SPEA2, Micro-GA, and NSGA-II. As verified for the Z_{CAP_1} modeling, recall that the negative values of power savings occurred at operational points whose fan speed in the collected data was 30 Hz. In these cases, the optimization also suggested increasing their speed in order to increase the tower's effectiveness, as already justified in Section 8.1. The consequent increase in the power consumption, which is coherent, matches the expected solution for the proposed optimization system.

Table 16 reports the values of the used metrics when the model based on Z_{CAP_2} was used. In the global evaluation of the algorithms using the Z_{CAP_2} modeling, the EEF was also used as the main metric for choosing the best algorithm. The values of EEF obtained are presented in Table 16. Recall that the APS (%) represents the energy savings obtained after performing the optimization; $\overline{\epsilon_a}$ represents the average effectiveness; AER (%) represents the difference between the average effectiveness obtained after applying the algorithms and the average effectiveness verified without optimization, both obtained by applying the cooling tower modeling to the field data. As before, we used the average effectiveness of 0.6761 as a reference, for the same reasons given in Section 8.1. Moreover, the reference

value for the computation of the effectiveness reduction after the optimizations had to be the same used for both modeling, since it was obtained not as a function of the modeling, but rather depending only on the cooling tower model, which was the same for all the scenarios compared in this work. The result is the possibility of a direct comparison of the global results presented by both models.

Table 16. Selection of the best algorithm for the Z_{CAP_2}-based model considering both stopping criteria.

Metric	After 50 Iterations					After 90 s				
	MOPSO	**MO-TRIBES**	**NSGA-II**	**SPEA2**	**MicroGA**	**MOPSO**	**MO-TRIBES**	**NSGA-II**	**SPEA2**	**MicroGA**
APS	9.35	9.19	8.28	8.48	8.43	9.48	9.27	8.32	8.50	8.43
$\overline{\epsilon_a}$	62.22	62.29	62.19	62.32	61.83	62.29	62.33	62.00	62.47	61.59
AER	5.39	5.32	5.42	5.29	5.78	5.32	5.28	5.61	5.14	6.02
EEF	1.73	1.73	1.53	1.60	1.46	1.78	1.76	1.48	1.65	1.40

It is noteworthy to point out that, in Table 16, for all algorithms, the value of EEF is greater than one, which is quite satisfactory. This means that the power savings achieved outweigh the reductions in effectiveness of the cooling tower. Figure 15 allows a visual comparison of the improvements yielded in average power savings for both stopping criteria (APS-50i and APS-90s) in relation to the reduction verified in the average effectiveness for both stopping criteria (AER-5i and AER-90s) of the tower for both applied algorithms.

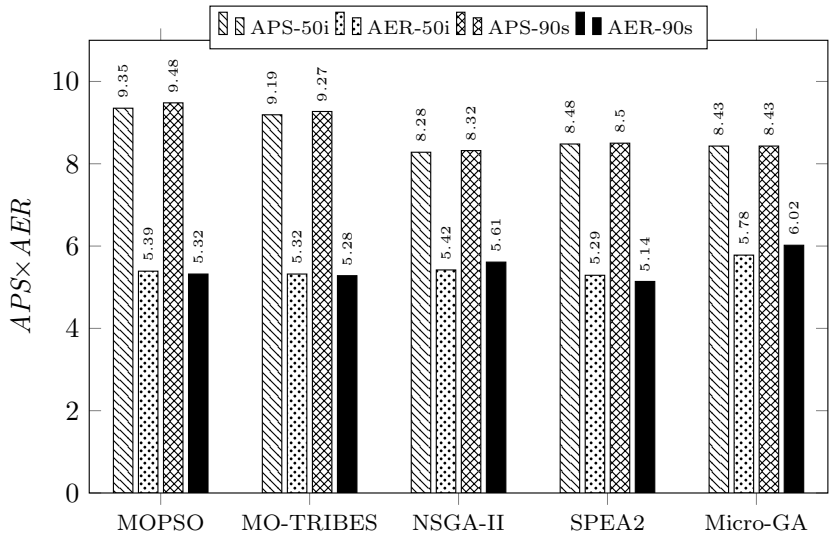

Figure 15. Comparison of average power savings and corresponding average effectiveness reductions regarding the Z_{CAP_2}-based model and considering both stopping criteria for the applied algorithms.

In this case, for the stopping criterion of 50 iterations, the MO-TRIBES and MOPSO algorithms show the same value of EFF, 1.73. These two algorithms had better performance than the other three algorithms. In decreasing order of performance, we have SPEA2, NSGA-II, and Micro-GA. After 90 s, it appears that MOPSO yielded the best result. The obtained power savings were about to 1.78 times the reduction in the tower effectiveness. In decreasing order regarding effectiveness, we have MO-TRIBES, SPEA2, NSGA-II, and then Micro-GA.

8.3. Model Comparison: Z_{CAP_1} vs. Z_{CAP_2}

Based on the comparison of the results obtained from the application of the models Z_{CAP_1} and Z_{CAP_2} for the compression chillers, we can select the best model for the application proposed in this work. Recall that the only advantage of using Z_{CAP_1} over the Z_{CAP_2}

is that the former uses a single search variable and the latter uses two variables. However, we verified that the additional variable used in Z_{CAP_2} did not impose a convergence time greater than 90 s, which is the maximum time period tolerated to run the required optimization process while the refrigeration system is online.

When using the Z_{CAP_1} modeling, we found out that for all five optimization algorithms, the reduction in effectiveness was greater than the increase in power savings, in contrast with what was verified when using the use of the Z_{CAP_2}-based model. This can be seen in the results reported in Tables 14 and 16, illustrated visually in Figures 14 and 15. Thus, the relationship between the power savings and the associated reduction in effectiveness obtained when using the Z_{CAP_2} modeling is synthesized by the energy efficiency factor (EEF). This evaluated to 1.78 against only 0.8072 obtained when using of Z_{CAP_1} modeling with both results yielded by MOPSO when applying the same stopping criterion of 90 s. In this context, MOPSO's performance is followed by that of MO-TRIBES, SPEA2, NSGA-II, and then Micro-GA. Considering EEF as a decisive metric for the performance analysis, the model based on Z_{CAP_2} exceeded by 2.21× the energy efficiency obtained by modeling Z_{CAP_1}. Therefore, the Z_{CAP_2} modeling was elected as the best chiller modeling for the proposed application. Figure 16 illustrates very well the superiority of the Z_{CAP_2}-based model over Z_{CAP_1} regarding the achieved energy efficiency (EEF) of the solutions found by the compared multi-objective algorithms.

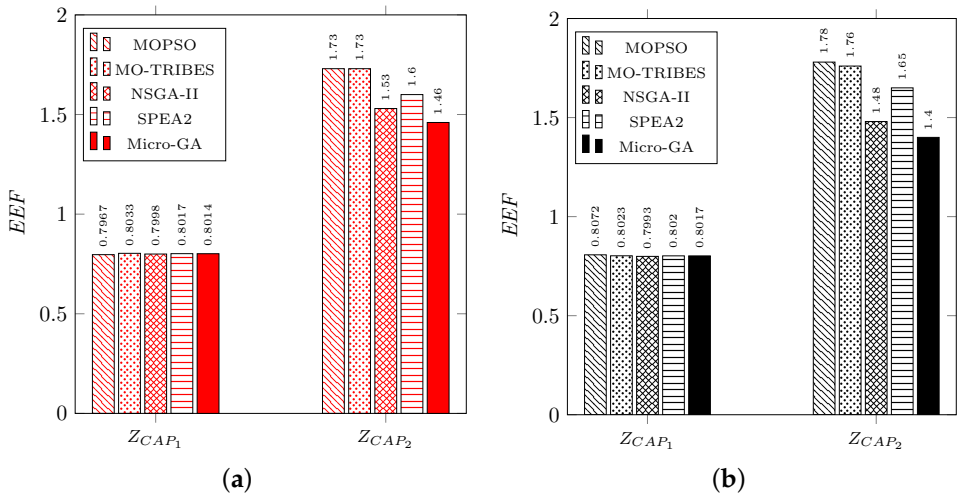

Figure 16. Comparison of the energy efficiency factors achieved by the compared algorithms for the performed optimizations: (**a**) 50 iterations; (**b**) 90 s.

9. Discussion

For this work, it was necessary to develop a mathematical model for the main equipment involved in the considered cooling process [4,5]. In addition, a preliminary survey was done in order to select the multi-objective optimization algorithms to be applied. Algorithms MOPSO and MO-TRIBES were thus elected to investigate their performances for approaching the optimization problem at hand. Their performances were compared to those of other multi-objective algorithms: NSGA-II, SPEA2, and Micro-GA.

The results of the application of the chosen algorithms were studied in terms of a fixed number of iterations and in terms of a time limit interval to obtain the optimal solution to be applied online to the real refrigeration system, hence yielding energy efficiency. The inferred iteration threshold was 50 iterations, and that of time was 90 s. These allowed us to meet the requirements of the application and verify and compare the performance impact. The comparison also aimed to verify the convergence of the algorithms, by comparing the results obtained using the two defined stopping criteria. After analyzing the obtained global performance results, we concluded that the results obtained with the stopping criterion of after 90 s should be adopted as a reference for evaluating the performances of

the algorithms. The MOPSO application outperformed that of MO-TRIBES, in terms of global results, after applying this criterion. It is noteworthy to emphasize that both applied algorithms yield better results than NSGA-II, SPEA2, and Micro-GA.

Two ways of modeling of the refrigeration system were used. Z_{CAP_1}, wherein the energy efficiency optimization was set to find the setpoint of only variable; and Z_{CAP_2}, wherein the optimization was set to find the setpoints of two variables. The MOPSO algorithm presented the best global performance for both models. In this case, even when MO-TRIBES executed a larger number of iterations than MOPSO for the same time interval, it was observed that, after 90 s, MOPSO obtained slightly superior performance. Considering the application of the Z_{CAP_1} modeling, in descending order of performance based on the established metrics, we had the following algorithm ranking: MOPSO, MO-TRIBES, SPEA2, Micro-GA, and NSGA-II. As for the Z_{CAP_2} modeling, also in decreasing order of performance, we had the following algorithm ranking: MOPSO, MO-TRIBES, SPEA2, NSGA-II, and Micro-GA.

The optimizations using the Z_{CAP_2} model performed better than those using the Z_{CAP_1} model. In this case, the main global performance metric considered was the energy efficiency factor (EEF). With the Z_{CAP_1} model, the EEF values were always lower than 1.0 for both applied algorithms, in contrast with those achieved when the Z_{CAP_2} model was used, for which the values of EEF were always greater than 1.0. This means that only using the modeling based on Z_{CAP_2} did the optimization process yield average global power savings that exceeded the reduction in the average effectiveness achieved by the tower. This result does not necessarily preclude the usage of the Z_{CAP_1} model, since the average effectiveness values obtained by both algorithms can be applied, in operational terms. For comparison purposes, the EEF obtained by MOPSO based on Z_{CAP_1} was about 0.81, and that based on Z_{CAP_2} was about 1.78, i.e., 2.21 times higher.

Considering the stopping criterion of 90 s, the application of the MOPSO algorithm using the Z_{CAP_2} modeling obtained average global power savings of 9.48%. This result is excellent. This result was obtained with an average tower effectiveness of 0.6229, which corresponds to a reduction of 5.32% in relation to the average effectiveness that would be obtained without performing the optimizations, which corresponds to 0.6761. The application of MOPSO using the Z_{CAP_1} yielded average global power savings of 5.01%, which corresponds to 56.11% of that obtained using the Z_{CAP_2}, and hence much less than that obtained using the Z_{CAP_2}. The latter result was obtained with an average effectiveness of 0.6141, corresponding to a 6.20% reduction in the average effectiveness that could be verified without performing the optimizations. This is a lower average effectiveness than that obtained with Z_{CAP_2}.

Still with regard to the Z_{CAP_1} modeling, the MO-TRIBES algorithm showed average energy savings of 5.27%, which is higher than that presented by MOPSO. However, MO-TRIBES obtained an average effectiveness of 0.6104, which is lower than that obtained by MOPSO, resulting in an EEF of 0.8023; compare that to 0.8072 obtained by MOPSO. Thus, although MO-TRIBES reached greater energy savings when using the Z_{CAP_1} model, in terms of global energy efficiency, and considering the EEF as the main performance evaluation metric of algorithms, MOPSO is considered the best algorithm. For both applied algorithms, the average speed of the tower's fans always remained above 50% of its nominal speed after the optimizations were carried out. This was the case for both models. The nominal value was determined as a practical limit for the application, due to the excessive consumption of water verified for lower speeds. In addition, the optimizations fully met the operational constraints established.

It is important to emphasize that the results obtained depend on the internal parameters of each algorithm, and mainly, on the decision criterion defined for choosing the optimal solution. The internal parameters of the algorithms were configured experimentally in order to obtain a Pareto front with good distribution during the defined time interval defined as the tolerated threshold for the application. It was observed that the decision criterion chosen for the application met the need to select an optimal solution

that would establish the best compromise between global electrical energy savings and the effectiveness of the cooling tower. It was also concluded that the interval of 90 s to obtain the solution to the problem is technically feasible, as it allows the proper convergence of the algorithms, and thereby good results. Moreover, the results obtained with the Z_{CAP_2} modeling show that the temperature variation of the chilled water leaving the chiller is allowed to be between 5.5° and 7.0° without impairing the thermal load or thermal comfort of the cooling system. Therefore, after analyzing all the scenarios presented in this work, we conclude that, for the proposed application, the optimization with the MOPSO algorithm using the Z_{CAP_2} modeling with a stopping criterion of 90 s represents the scenario that allowed obtaining the greatest possible energy efficiency for the considered cooling system. This allowed obtaining the greatest average savings in electrical energy accompanied by the smallest reduction in the average effectiveness of the cooling tower.

10. Conclusions

This work investigated the feasibility of multi-objective optimization based on swarm intelligence for the operation of cooling towers based on chillers, in order to obtain the operational setpoints that best compromise between two conflicting objectives: reducing the energy consumption and increasing the tower's effectiveness, aiming at the maximum energy efficiency possible for the refrigeration system. Two algorithms were applied: MOPSO and MO-TRIBES, for which we exploited two stopping criteria to select the most adequate for an online application of the optimization process. The first was 50 iterations, and the second, 90 s. Moreover, the usage of two system's models was analyzed: one based on a single search variable (Z_{CAP_1}) and the other on two search variables (Z_{CAP_2}). We compared the performances of the applied algorithms with those obtained by three other multi-objective algorithms, NSGA-II, SPEA2, and Micro-GA.

Based on the average power savings and the achieved energy efficiency of the system, we conclude that the best model is the one based on Z_{CAP_2} combined with the application of MOPSO. The selected stopping criterion was the one with fixed 90 s of optimization. The Discussion section brought forth the details of that conclusion. For the studied configuration, the aforementioned scenario achieved an energy efficiency factor of 1.78, allowing power savings of 9.48% with a tower effectiveness reduction of only 5.32%.

There are many possible ways to improve the presented work. There are several types of chillers which have models that depend on different variables from those used in this work. In addition, it is possible to compare the performances of the chosen algorithms by applying as an optimization strategy the variation of the speed of other equipment, such as condensed and chilled water pumps, and chillers. This can be approached through the application of frequency converters, similarly to the strategy adopted in this work. In this case, the variation in the speed of the cooling tower fans was taken into account. It is even possible to verify the feasibility of implementing the optimizations considering the speed variation in more than one of the mentioned pieces of equipment. In the present work, the increase in terms of water consumption of the cooling system was not estimated in relation to the reduction in the effectiveness of the cooling tower. Thus, developing a model that estimates this water consumption as a function of the effectiveness of the tower would be interesting. It would allow one to estimate the real gain in financial terms, comparing the reduction in electricity costs to the increase in expenses with the estimated increase in water consumption due to the reduction in the tower's effectiveness. Furthermore, the influence of the equipment that composes the secondary circuit, i.e., the chilled water circuit, which is composed of fan coils, was not considered. Thus, a deeper study would be possible by including the modeling of fan coils in the energy consumption of the cooling system. Thus, the implemented optimization system would also determine the optimal number of fan coils that should operate simultaneously to meet a certain established thermal comfort criterion, such as operation within a permissible range of ambient temperature. There is also the possibility to explore the usage of other kinds of multi-objective optimization

algorithms, such those based on evolutionary selection strategies as apposed to the swarm intelligence strategy. Among these algorithms, we can cite NSGA, Micro-GA, and SPEA.

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

Funding: This reaserch received funding from Fundação Carlos Chagas Filho de Amparo à Pesquisa do Estado do Rio de Janeiro (FAPERJ http://www.faperj.br) for funding this research (grant numbers 203.111/2018 and 750 201.013/2022).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data that support the findings of this study are available upon request. The data are not publicly available due to privacy or ethical restrictions.

Acknowledgments: We are grateful to the Fundação Carlos Chagas Filho de Amparo à Pesquisa do Estado do Rio de Janeiro (FAPERJ http://www.faperj.br) for funding this research (grant numbers 203.111/2018 and 750 201.013/2022). We also would like to thank Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq http://www.cnpq.br) for the continuous financial support.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Yu, K.; Hung, Y.; Hsieh, S.; Hung, S. Chiller Energy Saving Optimization Using Artificial Neural Networks. *J. Appl. Sci.* **2011**, *16*, 3008–3014. [CrossRef]
2. Lizarazu, M.S.D. Otimização Multiobjetivo Aplicada á Efciência Energética de Torres de Resfriamento. Master's Thesis, Universidade do Estado do Rio de Janeiro, Rio de Janeiro, Brazil, 1996. Available online: https://www.pel.uerj.br/bancodissertacoes/Dissertacao_Marcelo_Lizarazu.pdf (accessed on 15 September 2022).
3. Nedjah, N.; de Macedo Mourelle, L.; Lizarazu, M.S.D. Evolutionary Multi-Objective Optimization Applied to Industrial Refrigeration Systems for Energy Efficiency. *Energies* **2022**, *15*, 5575. [CrossRef]
4. Nedjah, N.; de Macedo Mourelle, L.; Lizarazu, M.S.D. Mathematical Modeling of Cooling Towers-Based Refrigeration Systems for Energy Efficiency Optimization. In Proceedings of the Computational Science and Its Applications—ICCSA 2022, Malaga, Spain, 4–7 July 2022; Springer International Publishing: Cham, Switzerland, 2022; pp. 289–307.
5. Nedjah, N.; de Macedo Mourelle, L.; Lizarazu, M.S.D. Mathematical Modeling of Chiller-Based Refrigeration Systems for Energy Efficiency Optimization. In Proceedings of the Computational Science and Its Applications—ICCSA 2022, Malaga, Spain, 4–7 July 2022; Springer International Publishing: Cham, Switzerland, 2022; pp. 273–288.
6. Alpina. *Torres de Resfriamento de Agua*; Technical Report; Alpina S/A Indústria e Comércio: São Bernardo do Campo, Brazil, 1978.
7. ASHRAE. *Handbook—HVAC Systems and Equipment*; SI Edition; American Society of Heating, Refrigerating and Air Conditioning Engineers: Peachtree Corners, GA, USA, 2012.
8. Li, X.; Li, Y.; Seem, J.E.; Li, P. Extremum seeking control of cooling tower for self-optimizing efficient operation of chilled water systems. In Proceedings of the 2012 American Control Conference, Montreal, BC, USA, 27–29 June 2012; pp. 3396–3401.
9. Liu, C.; Chuah, Y. A study on an optimal approach temperature control strategy of condensing water temperature for energy saving. *Int. J. Refrig.* **2011**, *34*, 816–823. [CrossRef]
10. Ma, Z.; Wang, S. Supervisory and optimal control of central chiller plants using simplified adaptive models and genetic algorithm. *Appl. Energy* **2011**, *88*, 198–211. [CrossRef]
11. Lu, L.; Cai, W.; Soh, Y.C.; Xie, L.; Li, S. HVAC system optimization—Condenser water loop. *Energy Convers. Manag.* **2004**, *45*, 613–630. [CrossRef]
12. Chow, T.T.; Zhang, G.Q.; Lin, Z.; Song, C.L. Global optimization of absorption chiller system by genetic algorithm and neural network. *Energy Build.* **2002**, *34*, 103–109. [CrossRef]
13. ASHRAE. *Principles of Heating, Ventilating and Air Conditioning*; American Society of Heating, Refrigerating and Air Conditioning Engineers: Peachtree Corners, GA, USA, 1998.
14. Haykin, S. *Redes Neurais: Princípios e Prática*, 2nd ed.; Bookman: Porto Alegre, Brazil, 2001.
15. Lee, K.; Cheng, T. A simulation-optimization approach for energy efficiency of chilled water system. *Energy Build.* **2012**, *54*, 290–296. [CrossRef]

16. Kusiak, A.; Li, M.; Tang, F. Modeling and optimization of HVAC energy consumption. *Appl. Energy* **2010**, *87*, 3092–3102. [CrossRef]
17. Kusiak, A.; Xu, G.; Tang, F. Optimization of an HVAC system with a strength multi-objective particle-swarm algorithm. *Energy* **2011**, *36*, 5935–5943. [CrossRef]
18. York. *Print-Out de Seleção das Unidades Resfriadoras UR-S2O3, UR-S2O4, UR-S2O5 e UR-S2O6*; Technical Report; YORK: Sorocaba, Brazil, 2006.
19. Alpina. *Memória de Cálculo das Torres de Resfriamento TR-2501 e TR-2502*; Technical Report; Alpina S/A Indústria e Comércio: São Bernardo do Campo, Brazil, 2013.
20. Coello, C.A.C.; Lechunga, M.S. MOPSO: A Proposal for Multiple Objective Particle Swarm Optimization. In Proceedings of the 2002 Congess on Evolutionary Computation, Part of the 2002 IEEE World Congress on Computational Intelligence, Honolulu, HI, USA, 12–17 May 2002; pp. 1051–1056.
21. Cooren, Y.; Clerc, M.; Siarry, P. MO-TRIBES, an Adaptive Multiobjective Particle Swarm Optimization Algorithm. *Comput. Optim. Appl.* **2011**, *49*, 379–400. [CrossRef]
22. York. *Curva de Surge das Unidades Resfriadoras UR-S2O3, UR-S2O4, UR-S2O5 e RU-S2O6*; Technical Report; YORK: Sorocaba, Brazil, 2009.
23. Merkel, F. *Verdunstungskuhlung*; VDI Forschungsarbeiten: Berlin, Germany, 1925; pp. 576–583.
24. Braun, J.E.; Klein, S.A.; Mitchel, J.W. Effectiveness models for cooling towers and cooling coils. *ASHRAE Trans.* **1989**, *95*, 164–174.
25. More, J.J. The Levenberg–Marquardt algorithm: Implementation and theory. *Numer. Anal.* **1978**, *630*, 105–116.
26. Zitzler, E.; Thiele, L. *An Evolutionary Algorithm for Multiobjective Optimization: The Strength Pareto Approach*; Technical Report 43; Computer Engineering and Communication Networks Lab (TIK), Swiss Federal Intitute of Technology (ETH): Zurich, Switzerland, 1999.
27. White, T.; Pagurek, B. Towards multi-swarm problem solving in networks. In Proceedings of the 3rd International Conference on Multi-Agent Systems (ICMAS 98), Paris, France, 3–7 July 1998; IEEE Computer Society: New York, NY, USA, 1998; pp. 333–340.
28. Bergh, F.v.d. An Analysis of Particle Swarm Optimizers. Ph.D. Thesis, Department of Computer Science, University of Pretoria, Pretoria, South Africa, 2001.
29. Millonas, M.M. *Swarms, Phase Transitions, and Collective Intelligence*; Addison Wesley: Reading, UK, 1994.
30. Eberhart, R.; Simpson, P.; Dobbins, R. *Computational Intelligence PC Tools*; Academic Press Professional: Boston, MA, USA, 1996.
31. Coelho, L.S.; Bernert, D.L.A. PID control design for chaotic synchronization using a tribes optimization approach. *Chaos Solitons Fractals* **2009**, *42*, 634–640. [CrossRef]
32. Moore, J.; Chapman, R. *Application of Particle Swarm to Multiobjective Optimization*; Department of Computer Science and Software Engineering, Auburn University: Auburn, AL, USA, 1999.
33. Deb, K. *Multi-Objective Optimization Using Evolucionary Algorithms*; Wiley and Sons: New York, NY, USA, 2001.
34. Hu, X.; Eberhart, R. Multiobjective Optimization Using Dynamic Neighborhood Particle Swarm Optimization. In Proceedings of the 2002 Congress on Evolutionary Computation, part of the 2002 IEEE World Congress on Computational Intelligence, Honolulu, HI, USA, 12–17 May 2002; pp. 1677–1681.
35. Coello, C.A.C.; Pulido, G.T.; Lechuga, M.S. Handling Multiple Objectives with Particle Swarm Optimization. *IEEE Trans. Evol. Comput.* **2004**, *8*, 256–279. [CrossRef]
36. Hu, X.; Eberhart, R. Particle Swarm Optimization with extended memory for multiojective optimization. In Proceedings of the 2003 IEEE Swarm Intelligence Symposion, Indianapolis, IN, USA, 24–26 April 2003; pp. 193–197.
37. Deb, K.; Goldberg, D.E. An investigation of niche and species formation in genetic function optimization. In Proceedings of the 3rd International Conference on Genetic Algorithms, Fairfax, VA, USA, 4–7 July 1989; Morgan Kaufmann Publishers Inc.: Burlington, MA, USA, 1989; pp. 42–50.
38. Mollazei, S.; Farsangi, M.M.; Nezamabadi-pour, H.; Lee, K.Y. Multi-objective Optimization of Power System Performance with TCSC Using the MOPSO Algorithm. In Proceedings of the Power Engineering Society General Meeting, Tampa, FL, USA, 24–28 June 2007; Volume 42; pp. 1–8.
39. Knowles, J.D.; Corne, D.W. Aproximating the Non-Dominated Front Using the Pareto Archived Evolution Strategy. *Evol. Comput. J.* **2000**, *8*, 149–172. [CrossRef] [PubMed]
40. Clerc, M. *TRIBES—Un Exemple d'Optimisation par Essaim Particulaire Sans Paramètres de Réglage*; Technical Report; Optimisation par Essaim Particulaire OEP 2003: Paris, France, 2003.
41. Cooren, Y.; Clerc, M.; Siarry, P. Initialization and Displacement of the Particles in TRIBES, a Parameter-Free Particle Swarm Optimization Algorithm. In *Adaptive and Multilevel Metaheuristics*; Cotta, C., Sevaux, M., Sorensen, K., Eds.; Studies in Computational Intelligence; Springer: Berlin/Heidelberg, Germany, 2008; Volume 136, pp. 199–219.
42. Cooren, Y.; Clerc, M.; Siarry, P. Tribes: A parameter-free particle swarm optimization. In Proceedings of the 7th EU Meeting on Adaptive, Self-Adaptive and Multi-Level Metaheuristics, Malaga, Spain, 16–17 November 2006; Vienna University of Technology: Vienna, Austria, 2006.
43. Chen, K.; Li, T.; Cao, T. Tribe-PSO: A novel global optimization algorithm and its application in molecular docking. *Chem. Intell. Lab. Syst.* **2006**, *82*, 248–259. [CrossRef]

44. Deb, K.; Agrawal, S.; Pratap, A.; Meyarivan, T. A Fast Elitist Non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In Proceedings of the Parallel Problem Solving from Nature VI Conference, Paris, France, 18–20 September 2000; Volume 1917, pp. 849–858.
45. Kukkonen, S.; Deb, K. Improved Pruning of Non-Dominated Solutions Based on Crowding Distance for Bi-Objective Optimization Problems. In Proceedings of the Evolutionary Computation, Vancouver, BC, Canada, 16–21 July 2006; pp. 1179–1186.
46. Sayyaadi, H.; Nejatolahi, M. Multi-objective optimization of a cooling tower assisted vapor compression refrigeration system. *Int. J. Refrig.* **2011**, *34*, 243–256. [CrossRef]
47. Liang, G.S. Fuzzy MCDM based on ideal and anti-ideal concepts. *Eur. J. Oper. Res.* **1999**, *112*, 682–691. [CrossRef]
48. INPE. Dados Observacionais do Centro de Previsão de Tempo e Estudos Climáticos. 2014. Available online: http://bancodedados.cptec.inpe.br (accessed on 15 September 2022).
49. Kie, P.L.T.; Theng, L. Intelligent Control of Heating, Ventilating and Air Conditioning Systems. In *Advances in Neuro-Information Processing*; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2009; Volume 5507, pp. 927–934.
50. Heris, S.M.K. Matlab Code for Multi-objective Particle Swarm Optimization (MOPSO)—Version 1.0. 2011. Available online: http://delta.cs.cinvestav.mx/~ccoello/EMOO/EMOOsoftware.html (accessed on 15 September 2022).
51. Clerc, M. TRIBES-D, A Fully Adaptive Parameter-free Particle Swarm Optimiser for Real Heterogeneous Problems. 2013. Available online: http://clerc.maurice.free.fr/pso/ (accessed on 15 September 2022).
52. Zitzler, E.; Thiele, L. Multiobjective Evolutionary Algorithms: A comparative case study and the strength pareto approach. *IEEE Trans. Evol. Comput.* **1999**, *4*, 257–271. [CrossRef]