agronomy

MDPI

Article

Metaheuristic Optimization of Agricultural Machinery for the Colombian Carnation Industry

Nixon Cuenca Orozco, Federico Gutiérrez Madrid * and Héctor Fabio Quintero

Mechanical Engineering, Applied Mechanics Department, Technological University of Pereira, Pereira 660003, Colombia; nixon.cuenca@utp.edu.co (N.C.O.); hquinte@utp.edu.co (H.F.Q.)
* Correspondence: fede9506@utp.edu.co

Abstract: The flower-growing sector in Latin America presents significant health risks for workers, which highlights the need for technological updates in their production processes. Likewise, outdated machinery leads to losses that need to be avoided. The method of productive innovation developed in this document involves optimizing a mechanism of agricultural machinery used in carnation classification. The optimization is achieved by minimizing the jerk of the mechanism's movement using metaheuristic methods. The results of three metaheuristic methods are compared against a brute force methodology. Optimization using these metaheuristic methods allows for achieving satisfactory results with up to 98% time reduction in the optimization process. This jerk optimization gives a longer useful life to the machinery, reduces the production stops needed for maintenance from once an hour to once every three hours, and reduces the damage done by the machine to the carnation stems.

Keywords: ant colony algorithm; flower industry; genetic algorithm; jerk minimization; mechanisms; sine cosine algorithm

check for updates

Citation: Cuenca Orozco, N.; Gutiérrez Madrid, F.; Quintero, H.F. Metaheuristic Optimization of Agricultural Machinery for the Colombian Carnation Industry. *Agronomy* **2024**, *14*, 2589. https://doi.org/10.3390/agronomy14112589

Academic Editors: Maofang Gao and Yuxing Han

Received: 3 October 2024
Revised: 23 October 2024
Accepted: 1 November 2024
Published: 3 November 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The advancement of society has meant that traditional manual management in agriculture is unable to meet demand. This makes the incorporation of modern and technological agricultural management essential for the sustainability and competitiveness of the sector. However, the high cost and complexity of adapting many of these innovations hinder their adoption. Furthermore, the shortage of professionals specialized in agricultural innovation and technology limits the quality of management and outcomes, highlighting the need for greater participation of engineering to meet the demands of the flower-growing sector.

The main players in the global export market of flowers had the following performance as of 2023: India, with 80.16% of the market (USD 12.4 billion); Colombia, with 6.78% (USD 1.1 billion); and Vietnam, with 5.50% (USD 854 million). Colombia's flower portfolio, by cultivated area, covers a total production of 77 km^2. The most cultivated flowers are roses, covering 26 km^2 (33.8% of the total area), and carnations, covering 9 km^2 (11.7%), representing a significant sector in the country's trade balance.

The incorporation of technologies in the agro-industrial sector has not been systematic because it is a heterogeneous area. This industry may require technological, process, logistical, or execution solutions that can vary depending on regional conditions and available resources [1]. Therefore, it is necessary to seek specific solutions to problems [2].

An additional problematic factor is that most of the available knowledge about each process comes from information networks that compare and complement traditional practices and the experiences of each worker. Moreover, in the case of Colombia, 85% of this formal knowledge comes from the personal expertise of workers [1]. Improvement opportunities, according to these information networks, do not consider the multiple technological solutions available in each sector because they lack systematic processes that encourage the inclusion of these new technologies.

An integrated perspective on technology and innovation highlights that innovations can vary in scale, from being new to a firm, to new in the market, or even new to the world. Innovation does not have to be global or transformative—small, localized changes are also recognized as innovations. This view rejects the notion that only high-tech or radical innovations matter, recognizing that innovation can arise in any sector. It also moves away from categorizing industries as "innovative" or "non-innovative", emphasizing that innovation is a continuous process. This perspective fits well with agriculture, where technological advancements emerge from both public and private efforts, and the industry itself is diverse, driven by unique technological and competitive dynamics.

At this point, it is worth noting that the incorporation of a technological contribution does not necessarily mean an immediate improvement in the process. In fact, manual work or a micro-innovation may be more efficient than a technological contribution due to the experience of the workers or the nature of the task. The truth is that many tasks (planting, harvesting, cutting, pest eradication, and classification) depend on the experience of the worker.

Due to the wide variety of flowers in the sector, technological contributions must be adapted to the specific type of flower, as each has unique characteristics and specific cultivation requirements. Moreover, the agro-industrial sector, especially floriculture, faces high occupational risks due to intensive manual labor, exposure to harmful chemicals, and the lack of regulation in the use of machinery. This leads to frequent muscular disorders, especially in the hands, wrists, and lower back. However, solutions for these problems have not been sufficiently studied, particularly in Colombia.

For the design of all flower industry machinery, there is a need to optimize to consider the lowest costs in the short and long term [3]. Over the years, various optimization methods have been developed, including programmed methods, heuristic algorithms, and metaheuristic algorithms aimed at optimizing the design process. Metaheuristic algorithms show the best performance, as they offer global solutions for design and avoid becoming trapped in local solutions, as is the case with heuristic algorithms [4].

Metaheuristic methods are designed to find the optimal solution to an optimization problem. They simulate behavioral patterns to define constraints and input values for each problem. In the design of mechanisms and machinery, they avoid falling into local solutions through phases of exploration and exploitation. Exploration means expanding the search space to new local areas, while exploitation involves finding the best local solution. The goal is to compare possible solutions and determine the optimal one [5]. These algorithms have three fundamental differences from conventional algorithms: they define a population in the search space, use specific fitness information, and employ probabilistic rules [6]. They can be used to solve non-differentiable or discontinuous optimization problems [7,8]. Due to their advantages, they can be applied in many disciplines within the field of general engineering [9,10], earth sciences [11,12], and, in this case, the flower-growing sector [13,14].

Recent research proposes modifying these methods, suggesting alternatives that improve their performance and replicability [15,16]. For example, the inclusion of tools and methodologies that diversify the possible solutions (target population) has been studied to avoid the premature convergence of the solution, aiming to escape from local optima [17].

The Salp Swarm Algorithm (SSA) is efficient in structural optimizations and has low complexity, but it struggles with cost, weight, and volume optimization, taking a long time to find the best solution [18]. The Multiverse Optimizer (MVO), while also facing challenges in geometry, weight, and volume optimization, is slow to converge and has high complexity [19]. The Moth-Flame Optimization Algorithm (MFO) offers fast convergence and low complexity, excelling in cost and weight optimization, but performs poorly in volume optimization [20]. Atom Search Optimization (ASO) has high computational complexity, leading to longer development and execution times [21]. Ultrasound-based optimization (EBO) performs well in optimizing weight, geometry, volume, and critical paths but is less effective in cost optimization and is both complex and time-consuming [22]. The Queue Search Algorithm (QSA) excels in optimizing weight, geometry, cost, and

volume, delivering quick results with low complexity [23]. The Equilibrium Optimizer (EO) performs well across weight, geometry, cost, and volume problems in short times, though it is highly complex [24].

Florval, a company located in Cundinamarca, Colombia, is grappling with urgent challenges related to the obsolescence of its mini carnation sorting machines. These machines incur high operating and maintenance costs and frequently fail, which significantly hampers productivity. In 2023, Florval processed over 67 million carnation stems annually using these machines. The high-impact movements of the machine's mechanism resulted in 6% of the stems being damaged, downgrading their quality and disqualifying them from the high-quality export classification. Additionally, the accumulation of broken stems necessitated hourly stops to clear waste buildup, risking total blockages and damage to mechanical components.

The primary objective of this study is to optimize the design of machinery used in the carnation industry, aiming to reduce mechanical damage to the stems, extend the equipment's lifespan, and minimize maintenance interruptions. Ultimately, this optimization seeks to increase the profitability of the agro-industry. To achieve this goal, three metaheuristic optimization methods are compared: Ant Colony Algorithm, Genetic Algorithm, and Sine Cosine Algorithm. These methods are employed to identify an optimal design configuration that minimizes jerk during the machine's operation. Additionally, the solutions obtained from these metaheuristic approaches are compared to those derived from an exhaustive search conducted over a reduced design space. This research aims to provide a robust solution that enhances machine performance while analyzing which algorithm demonstrates the best efficiency during the optimization process.

2. Materials and Methods

2.1. Mini Carnation Sorting Machine

Figure 1 displays the machine utilized for tying and sorting carnations at Florval S.A.S, located Nemocón, a municipality situated in the northern part of the Bogotá savanna in Colombia. Figure 2 illustrates a simplified representation of the mechanism to be optimized. These figures identify point O, the reference point of rotation; point A, the contact point with the cam of the mechanism; and point B, the tying point. The movement of point B facilitates the tying of carnation stems, and optimizing this movement is essential to reduce mechanical damage to the flowers.

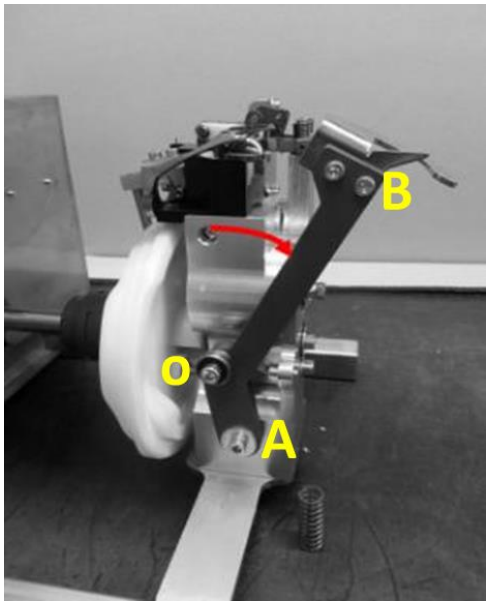


Figure 1. Carnation-tying machine mechanism, red arrow indicates movement direction.

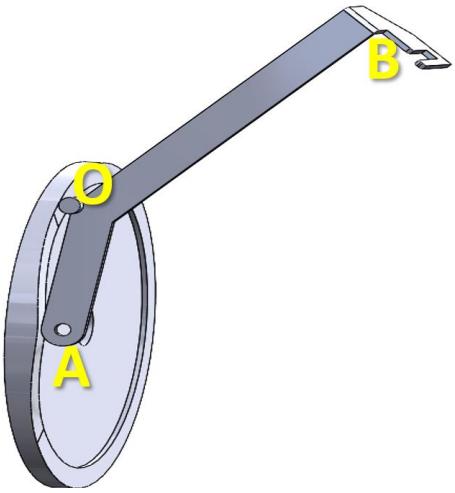


Figure 2. Carnation-tying machine mechanism simplification, rotation reference (point O), cam contact point (A), flower tying point (B).

An objective function was formulated specifically to minimize the jerk value for the tying point of the mechanism (point B). The jerk value represents the third derivative of the position with respect to time. To achieve this, it was developed the position equations for point B, considering its planar movement relative to the fixed reference point of rotation (point O). The position equations of X and Y directions, model the dynamic behavior of the mechanism as a function of time and are presented in Equations (1) and (2). These equations were derived based on the kinematic analysis of the mechanism, and they reflect the precise motion of the components involved in the optimization process.

$$x(t) = -L\cos\left(\arcsin\left(\frac{X_0 + 5\cos 8\pi t - 5}{r}\right) + \frac{3\pi}{4}\right) \tag{1}$$

$$y(t) = L\sin\left(\arcsin\left(\frac{X_0 + 5\cos 8\pi t - 5}{r}\right) + \frac{3\pi}{4}\right) \tag{2}$$

The design parameters L, r, and X_0 are the variables on which the optimization function depends. These represent physical measurements within the mechanism. Specifically, variable L represents the distance between points O and B, r denotes the distance between points O and A, and X_0 corresponds to the horizontal distance between the cam's stabilization section and point O. These variables are constrained to specific intervals due to the spatial limitations of the machine, which are detailed below.

- $L : [130, 146]$ mm;
- $r : [315, 380]$ mm;
- $X_0 : [25, 45]$ mm.

By calculating derivatives with respect to time using Equations (1) and (2) three times, the jerk functions are obtained. Due to its mathematical extension, equations of these derivatives are shown in detail in the simulation code upon request. However, their behavior is exemplified using arbitrary values L, r and X_0; then, the jerk vector magnitude is computed for a full turn of the cam which is shown in Figure 3. This figure shows the maximum jerk point in the plotted curve; the mechanism has 2 points of maximum jerk over every turn, which are the values being minimized through the optimization in this document. This computation is exemplified in Equations (3)–(5).

$$J_x(t) = \frac{d^3x(t)}{dt^3}, \tag{3}$$

$$J_y(t) = \frac{d^3y(t)}{dt^3}, \tag{4}$$

$$|J(t)| = \sqrt{J_x^2 + J_y^2}. \quad (5)$$

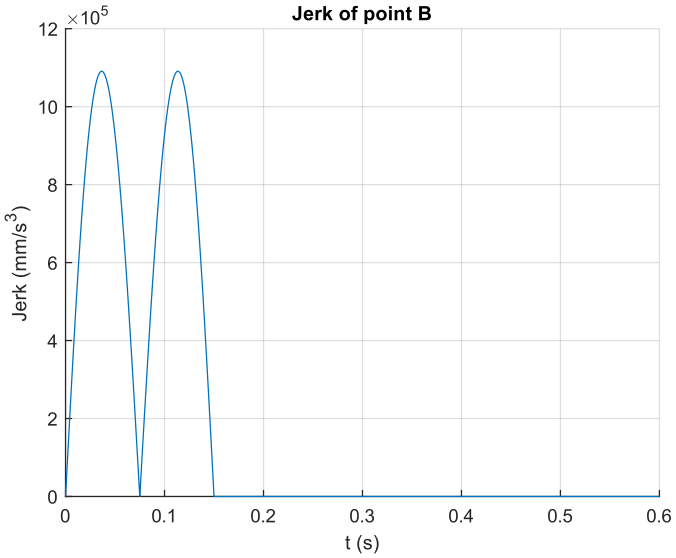


Figure 3. Jerk as a function of time for a full cam turn.

2.2. Brute Force Method

The brute force method for optimization is a straightforward approach that involves evaluating every possible solution within a defined search space to identify the best outcome. This method systematically explores all configurations, calculating the objective function for each one and ensuring that no potential solution is overlooked. While the brute force method guarantees finding the optimal solution, it can be computationally expensive and time-consuming, especially for complex problems with large search spaces. As a result, it is often impractical for real-world applications where efficiency is crucial. Despite its limitations, the brute force method serves as a valuable baseline, providing a clear benchmark against which the performance of more sophisticated optimization techniques can be compared.

The search space for the current optimization consisted of a millimeter-by-millimeter combination of the values of the variables within the specified intervals. For variable L, the range was from 130 mm to 146 mm, providing 17 discrete values. Variable r ranged from 315 mm to 380 mm, resulting in 66 possible values. Lastly, variable X_0 was defined within the interval of 25 mm to 45 mm, offering 21 values. In total, these parameters created 23,562 combinations to be evaluated.

2.3. Sine Cosine Algorithm

The Sine Cosine Algorithm (SCA) is a metaheuristic search methodology based on a random sinusoidal scale, from the optimal point found up to that iteration. This search methodology allows for a balance between exploration and exploitation. The algorithm demonstrates great potential for global optimization and shows promising results when modified or hybridized with other algorithms, particularly in solving design problems [25–27]. Its simplicity allows for a wide range of applications and is governed by Equations (6)–(9) [28].

$$r_1 = b - b\left(\frac{i}{T}\right), \quad (6)$$

$$r_2 = 2\pi rand(), \quad (7)$$

$$r_3 = 2rand(), \quad (8)$$

$$X_{i+1} = X_i + r_1 sin(r_2)|r_3 X_m - X_i|, \quad (9)$$

- r_1: Attenuation parameter;
- r_2, r_3: Random numbers;
- b: Search intensity parameter;
- T: Number of iterations;
- i: Current interaction;
- X: Variables to optimize;
- X_m: Value of the optimal variables found according to the optimization criterion.

The parameter b and the number of iterations T are critical in implementing the method. The parameter b, which controls search intensity, defines the amplitude of the sine function, and governs how the search is conducted around the optimal point. This variable indicates the maximum value that the attenuation parameter can reach. When b is larger, the search explores a wider region of the optimization space, enhancing exploration. However, this may reduce exploitation, as the optimization steps become larger. To balance exploration and exploitation, it is necessary to correlate b with the number of iterations T.

In Figure 4, the behavior of the maximum jerk in relation to b is shown for a constant number of iterations. The chosen interval for b is [0.01, 0.1], with 100 values of b evaluated within this range. The number of iterations T is set to 1000. The computations were repeated 100 times to calculate the mean and standard deviation for each of the 100 b values. The results indicate that when b increases, the optimization tends to stabilize at a constant value. Conversely, when b is too small, the exploration becomes less effective, leading to significant variability in the optimal value, as reflected by a higher standard deviation. This suggests that, for this number of iterations, a b value greater than approximately 0.05 tends to achieve a lower maximum jerk value more consistently.

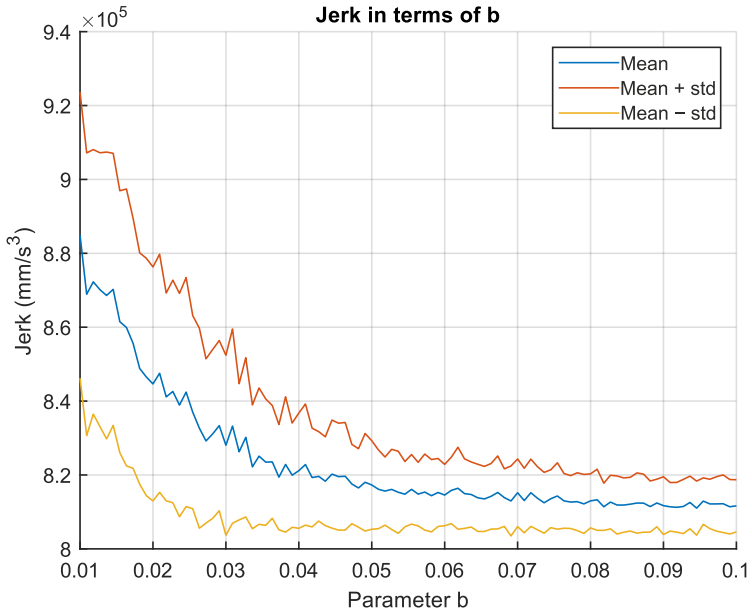


Figure 4. Jerk as a function of the search intensity parameter (b) for a constant value of the number of iterations (T).

In Figure 5, the behavior of the maximum jerk is depicted as a function of the number of iterations for a constant parameter b. The chosen interval for the number of iterations is [10, 1000], with 100 iteration values sampled within this range. The parameter b is set to 0.1 for this computation. The process was repeated 100 times to calculate the mean and standard deviation for each of the 100 iteration values.

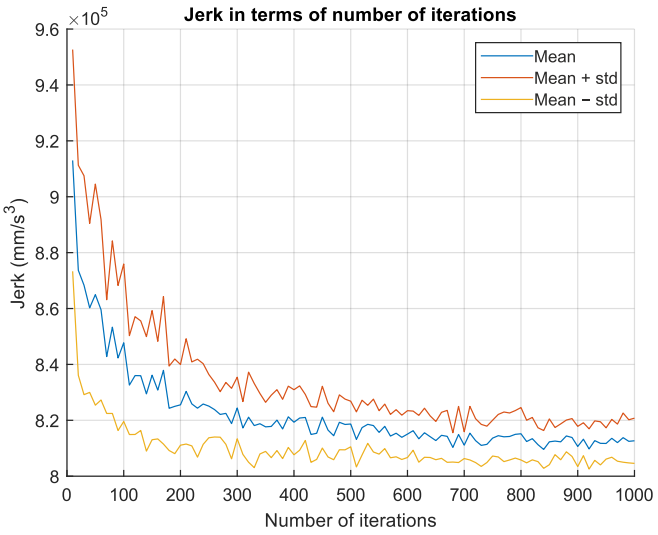


Figure 5. Jerk as a function of the number of iterations (T) for a constant value of the search intensity parameter (b).

This figure shows that increasing the number of iterations T leads to more optimal solutions. For this value of b, the jerk values tend to stabilize around 500 iterations, with a noticeably lower standard deviation. This indicates that after a certain threshold, additional iterations yield diminishing returns in terms of optimization improvement but contribute to consistency.

The parameter selection for the Sine Cosine Algorithm (SCA) in the jerk optimization process was based on the results shown in Figures 4 and 5. Values of $b = 0.1$ and $T = 1000$ iterations were chosen for the optimization.

2.4. Particle Swarm Algorithm

Particle Swarm Optimization (PSO), introduced by Eberhart and Kennedy, is inspired by human social behavior and natural swarming phenomena, such as fish schools or bird flocks [29]. In PSO, each solution is represented by a particle, with its position reflecting a potential solution, while the 'food' is the objective function being optimized. Throughout the iterations, particles adjust their positions by moving toward both the best local and global solutions, eventually converging on the optimal objective value. The main steps in PSO include initializing parameters, evaluating the objective function, and updating the particles' velocity and position.

In recent advancements in trajectory planning for high-speed parallel manipulators, Chen et al. [30] presented an innovative technique that optimizes execution time, energy consumption, and jerk, utilizing a quantum-behaved particle swarm optimization (QPSO) algorithm and quintic B-spline interpolation to enhance efficiency, reduce vibration, and minimize energy use. Xiao et al. [31] proposed a jerk-limited heuristic feed rate scheduling method for five degrees-of-freedom (5-DOF) hybrid robots, optimizing machining time and stability using particle swarm optimization (PSO). Their approach addresses the complexity of feed rate scheduling for high-precision machining, improving both computation speed and overall efficiency.

In the realm of product design, Fu et al. [32] propose an innovative method that integrates Kansei Engineering with Game Theory and Particle Swarm Optimization–Support Vector Regression (PSO-SVR) to enhance the objectivity and precision of designing Ming-style furniture, addressing key emotional requirements from users.

In this document, it is applied a methodology derived from the Particle Swarm Optimization (PSO) algorithm, specifically the Ant Colony Optimization (ACO) algorithm. The classical ACO requires the definition of several key parameters for effective operation. These parameters include the following:

- Number of ants that are going to choose each path.
- Evaporation coefficient.

In the Ant Colony Optimization approach for solving a continuous function, the problem is addressed by discretizing the variables into intervals. For the current problem, which involves minimizing the jerk, three variables need to be optimized. The different paths represent the combinatorial space formed by the discretization of the continuous intervals, and the reference points along these paths correspond to the potential values of each variable.

To define these paths, it is essential to establish the optimization order for the variables. In this case, as illustrated in Figure 6, the variables are optimized in the following order: first r, then L, and finally X_0. Each white point in this figure represents a possible value for one of the variables. The path chosen by an ant is a combination of one white point from each variable circle. The ACO implementation used 10 ants to explore the solution space.

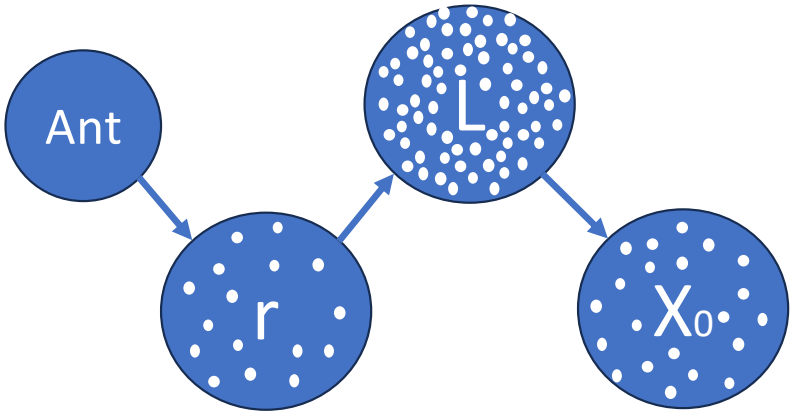


Figure 6. Diagram of Ant Colony Optimization path implemented.

The evaporation coefficient, which determines how quickly undesirable paths are removed from the ants' preferences, was set to 0.05 in this solution. Additionally, a stopping criterion of five iterations without improvement in the optimal value was employed.

2.5. Genetic Algorithm

Genetic Algorithm (GA) was proposed by Holland John [33] and has been widely employed to select optimal features, especially in the development of envelop feature selection methods [34]. It is considered as one of the first and most widely applied metaheuristic algorithms. The design inspiration for GA comes from natural selection theory and genetic principles. The fundamental idea is to simulate the process of biological evolution, evolving individuals within a population over generations through genetic operations. This iterative evolution aims to produce individuals increasingly adapted to the environment, searching for the optimal solution.

Decision variables are considered chromosomes when GA is employed to address optimization problems. These chromosomes are updated according to the genetic principles of genes and undergo operations such as crossover and mutation. Subsequently, based on the values of their objective function, the best chromosomes are preserved through survival of the fittest. After multiple iterations, the chromosome satisfying the termination condition is regarded as the result of model parameter inversion. In their study on air conditioning systems, Zhao et al. [35] proposed an innovative angle control algorithm for air curtains using a GA-optimized quadratic BP neural network, achieving improved airtightness and real-time performance, which is crucial for enhancing energy efficiency and emission reduction. Tegegne and Kim [36] used Genetic Algorithms to develop and evaluate the operation rules of a reservoir reflecting the uncertainty of the reservoir inputs.

The classical Genetic Algorithm implemented in this document uses a crossover mechanism where one offspring inherits a single chromosome from the father and two

chromosomes from the mother, or vice versa. The process then applies an elitist selection, choosing the better solution from the two offspring. In this case, each chromosome represents one of the three variable values, or genes, selected for the optimization process. After crossover, a random mutation is applied to one of the chromosomes of the elite offspring by altering neighboring genes, which helps introduce diversity to the population of ten individuals. The stopping criterion is set at 50 generations without improvement in the optimization value.

The discretization of possible genes is achieved by initializing variables and ensuring that mutations can only occur within predefined intervals for each variable. This approach guarantees that any geometrical value accepted as optimal remains within the defined range, and the most suitable values are represented among the genes in the population across generations. This method helps maintain the precision of the optimization process while ensuring that the search space is properly explored within the allowed boundaries.

3. Results

Using a brute force methodology, the minimum possible jerk value among all maximum jerk values was obtained, as well as the values of the design variables. The absolute optimal values found in the search space were the followings:

- $r = 146$ mm;
- $L = 315$ mm;
- $X_0 = 25$ mm.

Figure 7 shows the value of the maximum jerk for all the values evaluated in the brute force method. A global minimum value of jerk is observed that is expected to be reached by means of the metaheuristic methods.

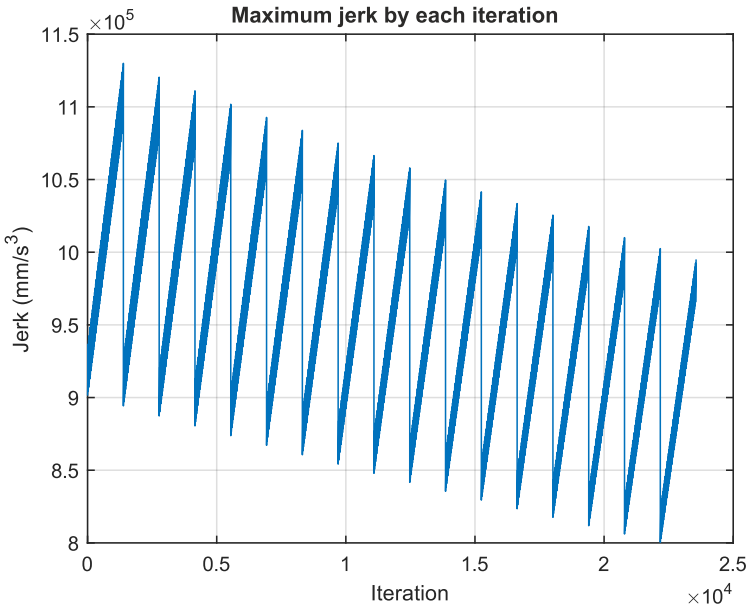


Figure 7. Maximum jerk as a function of iteration for the brute force method.

From the behavior shown in Figure 7, 17 trend peaks are observed as the variable values change, corresponding to the 17 values of the variable L. The jerk values in Figure 7 range between 112,890 N/mm^3 and 80,049 N/mm^3. This indicates that there could be a 29% reduction in jerk according to the change from the maximum jerk value (112,890) to the optimized jerk value (80,049), which was calculated using Equation (10).

$$\left(\frac{112,890 - 80,049}{112,890}\right) \cdot 100 = 29\% \tag{10}$$

Figure 8 shows the graph obtained from the maximum jerk values as a function of iteration for the Ant Colony Algorithm. It is observed that the algorithm reached the optimal value in iteration 4. This shows high efficiency in finding the best solution.

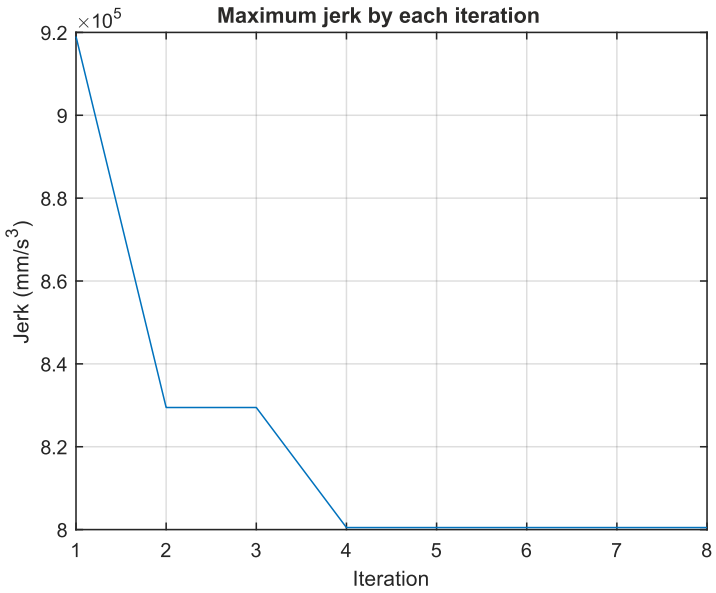


Figure 8. Maximum jerk as a function of iteration for the Ant Colony Algorithm.

Figure 9 illustrates the graph of maximum jerk values as a function of iteration for the Genetic Algorithm. The optimal value is reached at iteration 52. Due to the inherent nature of Genetic Algorithms, multiple iterations, or generations, are typically required to converge on the best solution. It is noteworthy that while the algorithm quickly achieves an acceptable level of optimization in the early iterations, reaching the true optimal point needs further refinement of the chromosomes across several generations.

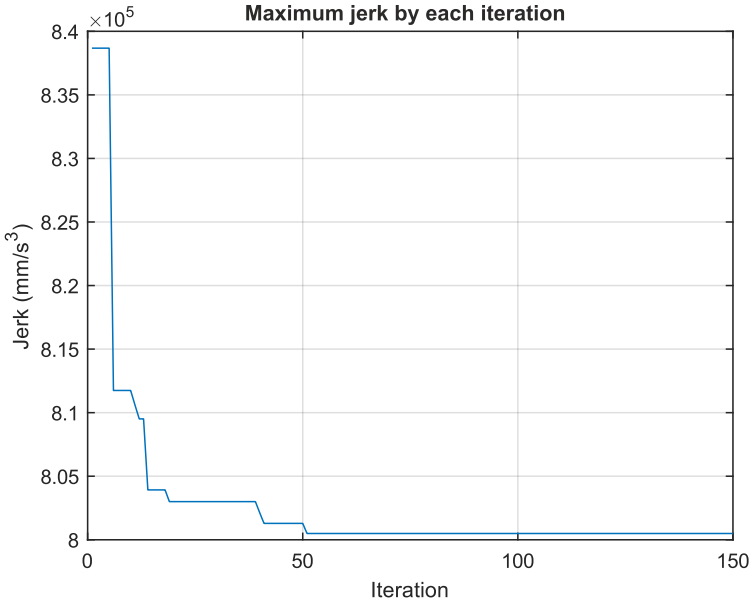


Figure 9. Maximum jerk as a function of iteration for the Genetic Algorithm.

Figure 10 shows the results obtained from the maximum jerk values as a function of iteration for the Sine Cosine Algorithm. It is observed that the algorithm reached the optimal value in iteration 491.

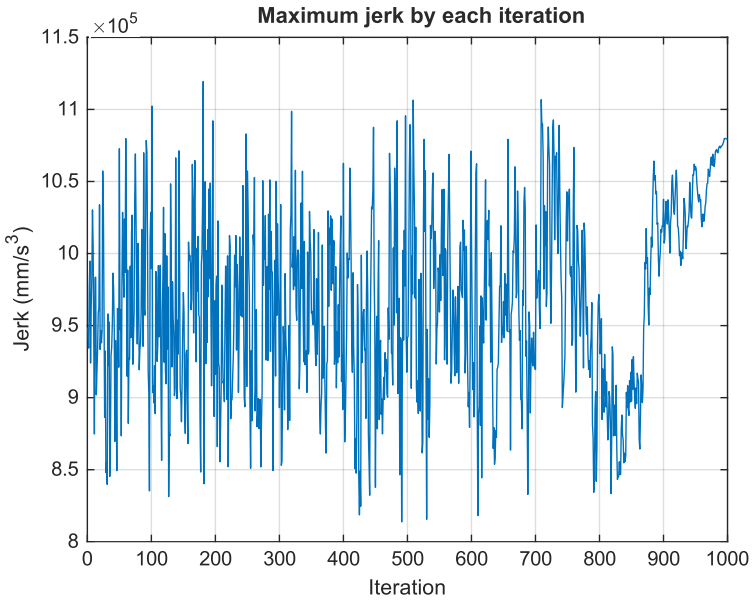


Figure 10. Maximum jerk as a function of iteration for the Sine Cosine Algorithm.

The results obtained using both the Genetic Algorithm and the Ant Colony Algorithm matched those from the brute force method but were achieved in significantly shorter times. Table 1 presents a comparison of the relative and absolute errors between the SCA method and the brute force method. This comparison is essential because SCA uses a continuous search approach, unlike the discrete methodology applied in the brute force method.

Table 1. Comparison of results of SCA versus brute force method.

Brute Force	Sine Cosine Algorithm	Absolute Error	Relative Error
$r = 146$ mm	$r = 145.91$ mm	0.09 mm	0.062%
$L = 315$ mm	$L = 315.71$ mm	0.71 mm	0.225%
$X_0 = 25$ mm	$X_0 = 26.75$ mm	1.75 mm	7%
$J_{max} = 80,049\ \frac{\text{mm}}{\text{s}^3}$	$J_{max} = 80,421\ \frac{\text{mm}}{\text{s}^3}$	$372\ \frac{\text{mm}}{\text{s}^3}$	0.465%

SCA: Sine Cosine Algorithm; r : Distance between points O and A; L : Distance between points O and B; X_0 : Horizontal distance between point O and the cam's stabilization point; J_{max}: Maximum jerk in the movement of the mechanism.

It can be observed that the relative error of X_0 was 7%, corresponding to an absolute value of 1.75 mm, which is the highest error among all variables. However, it is important to note that the relative error of the objective value, J_{max}, was only 0.465%. This indicates that while X_0 has a larger margin of error, it does not significantly influence the optimization of the jerk value. Therefore, the 7% error for X_0 can be considered acceptable within the context of the overall optimization.

Figure 11 presents a comparison of the execution times for each algorithm, averaged over 10 random runs. On one hand, both the Ant Colony and Genetic Algorithms consistently find the optimal solution in each run. On the other hand, while the Sine Cosine Algorithm yields varying results across runs, the variations are not significant.

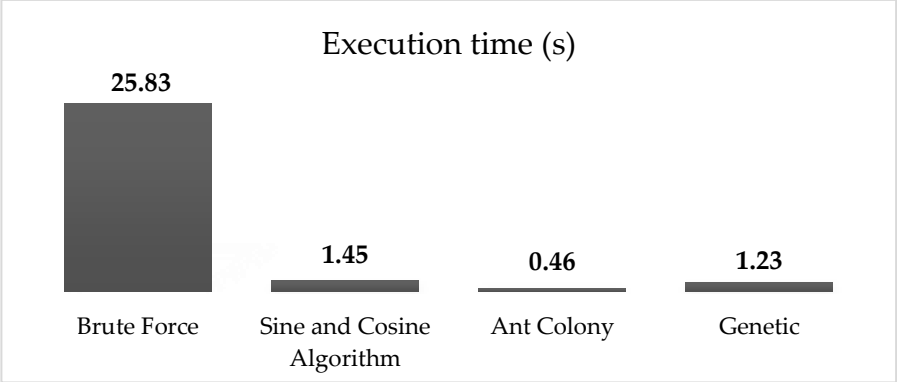


Figure 11. Comparison of execution time between all optimization methods.

The reduction in execution time with respect to the brute force method was computed using Equation (11).

$$\left(\frac{\text{Brute force's execution time} - \text{Other algortihm's execution time}}{\text{Brute force's execution time}}\right) \cdot 100 \quad (11)$$

With this equation, a reduction of 94% in execution time was obtained for the Sine Cosine and Genetic Algorithms. Also, a reduction of 98% was obtained for the Ant Colony Algorithm.

4. Discussion

The sinusoidal random search characteristic of the Sine Cosine Algorithm (SCA) typically allows for finding an optimal function value within just a few iterations. However, in this instance, it took 491 iterations to arrive at the result, which is approximately half of the predetermined maximum iterations. The stopping criterion for this method was set to the maximum iterations to ensure that the algorithm could continue searching for an absolute minimum value around the optimal point that had already been identified. Since the search occurs within a continuous interval rather than a discrete set of possibilities, it is expected that the Sine Cosine Algorithm (SCA) may not achieve the absolute minimum identified by the brute force method. The constraints on the possible values due to interval restrictions limit the algorithm's sinusoidal nature. This can negatively impact the algorithm's random search capability, leading it to converge on a local minimum from which it does not improve, even after completing all iterations. Nonetheless, the absolute and relative errors in the results are small, as shown in Table 1. Notably, the maximum jerk value obtained by both methods differs by only 0.465%, indicating that the difference is not significant and SCA could be an effective option in the optimization process.

On the other hand, the Genetic Algorithm and the Ant Colony Algorithm achieved a value equal to the one determined by brute force. This means that accuracy it is not a problem for these algorithms when a correct discretization of possible solutions in the search space is implemented.

The difference in processing time between the different metaheuristic methods is not higher than 1 s; however, in all cases the execution time is less than 6% of the time needed by the brute force method, which is highly desirable.

The Ant Colony Algorithm is the most efficient metaheuristic algorithm for this optimization, as both the processing time and the number of internal iterations of the algorithm are the smallest among all those used.

The 29% reduction in jerk values achieved through the dimensional optimization of the mechanism has a significant positive impact on the daily handling of carnation stems. After implementing physical modifications to the machine based on the optimal values, Florval Company observed a decrease in mechanical damage to the stems from 6% to 2%, according to the monthly production summary. Furthermore, the frequency of

hourly maintenance was extended to once every three hours, minimizing downtime on the production line. This optimization not only enhances operational efficiency but also reduces mechanical wear, contributing to a longer lifespan for the equipment.

5. Conclusions

The optimization of the construction variables of a mechanism found in machinery for sorting and processing carnations in the Colombian flower industry was carried out. This optimization was carried out by means of an exhaustive search known as brute force, comparing it with three metaheuristic algorithms, a Sine Cosine Algorithm, an Ant Colony Algorithm, and a Genetic Algorithm. Through this optimization of geometric parameters, it was possible to reduce the impact generated by the repetitive movement of the mechanism, represented by the maximum value of the jerk during its regular path. At the same time, this reduced the damage done to the carnation stems and mechanical wear. By means of these metaheuristic methodologies, results can be obtained in considerably shorter times with a precision that does not have a negative effect to consider in the general optimization of the mechanism. Finally, the Ant Colony Algorithm demonstrates superior performance among the analyzed metaheuristic methods for dimensional optimization, as evidenced by its shorter execution times and the optimal values achieved.

Author Contributions: All authors contributed to this study's conception and design. The conceptualization and methodology definition were performed by N.C.O. and H.F.Q. Material preparation, data collection, and analysis were performed by N.C.O. and F.G.M. The implementation of the computer code and supporting algorithms was performed by F.G.M. The first draft of the manuscript was written by N.C.O. and F.G.M. The manuscript was revised and polished by H.F.Q. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The datasets used and/or analyzed during the current study as well as the algorithm and support code used are available from the corresponding author upon reasonable request.

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

References

1. International Trade Center (ITC). Trade Statistics for International Business Development Monthly, Quarterly, and Yearly Trade Data. Import & Export Values, Volumes, Growth Rates, Market Shares, 2024. Available online: https://www.trademap.org/stDataAvailability.aspx (accessed on 9 July 2024).
2. International Association of Horticultural Producers (AIPH). AIPH's International Vision Project (IVP) Sets out to Identify the Future Drivers and Inhibitors That Influence Global Ornamental Horticultural Production, 2024. Available online: https://aiph.org/giic/international-vision-project (accessed on 10 July 2024).
3. Ullah, A.; Asami, K.; Holtz, L.; Röver, T.; Azher, K.; Bartsch, K.; Emmelmann, C. A Machine Learning Approach for Mechanical Component Design Based on Topology Optimization Considering the Restrictions of Additive Manufacturing. *J. Manuf. Mater. Process.* **2024**, *8*, 220. [CrossRef]
4. Yu, L.; Qin, H.; Xiang, P. Incentive mechanism of different agricultural models to agricultural technology information management system. *Sustain. Comput. Inform. Syst.* **2020**, *28*, 100423. [CrossRef]
5. Lalaymia, H.; Djellad, A.; Rekik, B.; Farou, M. Revolutionizing PV grid integration: Metaheuristic optimization of fractional PI controllers in T-type neutral point piloted inverters for enhanced performance. *Comput. Electr. Eng.* **2024**, *120*, 109694. [CrossRef]
6. Xu, H.; Guo, D.; Lu, J.; Niu, X. SWOT Analysis on the Development of Modern Ecological Landscape Agriculture: A Case Study of Baiquan Ecological Park. *Asian Agric. Res.* **2016**, *8*, 12–20. [CrossRef]
7. Agrahari, A.; Dhabu, M.M.; Deshpande, P.S.; Tiwari, A.; Baig, M.A.; Sawarkar, A.D. Artificial Intelligence-Based Adaptive Traffic Signal Control System: A Comprehensive Review. *Electronics* **2024**, *13*, 3875. [CrossRef]
8. Mayouf, C.; Salhi, A.; Haidara, F.; Aroua, F.Z.; El-Sehiemy, R.A.; Naimi, D.; Aya, C.; Kane, C.S.E. Solving Optimal Power Flow Using New Efficient Hybrid Jellyfish Search and Moth Flame Optimization Algorithms. *Algorithms* **2024**, *17*, 438. [CrossRef]
9. Ma, R.; Karimzadeh, M.; Ghabussi, A. Assessment of composite beam performance using GWO-ELM metaheuristic algorithm. *Eng. Comput.* **2022**, *38*, 2083–2099. [CrossRef]
10. Karim, F.K.; Khafaga, D.S.; Eid, M.M.; Towfek, S.K.; Alkahtani, H.K. A Novel Bio-Inspired Optimization Algorithm Design for Wind Power Engineering Applications Time-Series Forecasting. *Biomimetics* **2023**, *8*, 321. [CrossRef]

11. Ruiz-Vélez, A.; Alcalá, J.; Yepes, V. A Parametric Study of Optimum Road Modular Hinged Frames by Hybrid Metaheuristics. *Materials* **2023**, *16*, 931. [CrossRef]
12. Debiche, F.; Benbouras, M.A.; Petrisor, A.-I.; Baba Ali, L.M.; Leghouchi, A. Advancing Landslide Susceptibility Mapping in the Medea Region Using a Hybrid Metaheuristic ANFIS Approach. *Land* **2024**, *13*, 889. [CrossRef]
13. Sabzi, S.; Abbaspour-Gilandeh, Y.; Garcia-Mateos, G. A fast and accurate expert system for weed identification in potato crops using metaheuristic algorithms. *Comput. Ind.* **2018**, *98*, 80–89. [CrossRef]
14. Shayanfar, H.; Gharehchopogh, F.S. Farmland fertility: A new metaheuristic algorithm for solving continuous optimization problems. *Appl. Soft Comput.* **2018**, *71*, 728–746. [CrossRef]
15. Zambou, M.C.Z.; Kammogne, A.S.T.; Siewe, M.S.; Azar, A.T.; Ahmed, S.; Hameed, I.A. Optimized Nonlinear PID Control for Maximum Power Point Tracking in PV Systems Using Particle Swarm Optimization. *Math. Comput. Appl.* **2024**, *29*, 88. [CrossRef]
16. Radcliffe, C.; Parissi, C.; Raman, A. Valuing Indigenous Knowledge in the Highlands of Papua New Guinea: A Model for Agricultural and Environmental Education. *Aust. J. Environ. Educ.* **2016**, *32*, 243–259. [CrossRef]
17. Wang, W.; Tian, W.; Xu, D.; Zang, H. Arctic puffin optimization: A bio-inspired metaheuristic algorithm for solving engineering design optimization. *Adv. Eng. Softw.* **2024**, *195*, 103694. [CrossRef]
18. Mirjalili, S.; Gandomi, A.H.; Mirjalili, S.Z.; Saremi, S.; Faris, H.; Mirjalili, S.M. Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems. *Adv. Eng. Softw.* **2017**, *114*, 163–191. [CrossRef]
19. Mirjalili, S.; Mirjalili, S.M.; Hatamlou, A. Multi-Verse Optimizer: A nature-inspired algorithm for global optimization. *Neural Comput. Appl.* **2016**, *27*, 495–513. [CrossRef]
20. Mirjalili, S. Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm. *Knowl.-Based Syst.* **2015**, *89*, 228–249. [CrossRef]
21. Zhao, W.; Wang, L.; Zhang, Z. A novel atom search optimization for dispersion coefficient estimation in groundwater. *Future Gener. Comput. Syst.* **2019**, *91*, 601–610. [CrossRef]
22. Nhu, V.H. Levenberg-Marquardt method for ill-posed inverse problems with possibly non-smooth forward mappings between Banach spaces. *Inverse Probl.* **2022**, *38*, 015007. [CrossRef]
23. Zheng, Y.J.; Ling, H.F.; Xue, J.Y. Ecogeography-based optimization: Enhancing biogeography-based optimization with ecogeographic barriers and differentiations. *Comput. Oper. Res.* **2014**, *50*, 115–127. [CrossRef]
24. Faramarzi, A.; Heidarinejad, M.; Stephens, B.; Mirjalili, S. Equilibrium optimizer: A novel optimization algorithm. *Knowl.-Based Syst.* **2020**, *191*, 105190. [CrossRef]
25. Li, Y.; Zhao, Y.; Liu, J. Dimension by dimension dynamic sine cosine algorithm for global optimization problems. *Appl. Soft Comput.* **2021**, *98*, 106933. [CrossRef]
26. Wang, M.; Lu, G. A Modified Sine Cosine Algorithm for Solving Optimization Problems. *IEEE Access* **2021**, *9*, 27434–27450. [CrossRef]
27. Singh, N.; Kaur, J. Hybridizing Sine–Cosine Algorithm with Harmony Search Strategy for Optimization Design Problems. *Soft Comput.* **2021**, *25*, 11053–11075. [CrossRef]
28. Bansal, J.C.; Bajpai, P.; Rawat, A.; Nagar, A.K. *Sine Cosine Algorithm for Optimization*; Springer: Singapore, 2023; pp. 24–30. [CrossRef]
29. Eberhart, R.; Kennedy, J. Particle swarm optimization. *Proc. IEEE Int. Conf. Neural Netw.* **1995**, *4*, 488968. [CrossRef]
30. Chen, W.; Wang, H.; Liu, Z.; Jiang, K. Time-energy-jerk optimal trajectory planning for high-speed parallel manipulator based on quantum-behaved particle swarm optimization algorithm and quintic B-spline. *Eng. Appl. Artif. Intell.* **2023**, *126*, 107223. [CrossRef]
31. Xiao, J.; Liu, S.; Liu, H.; Wang, M.; Li, G.; Wang, Y. A jerk-limited heuristic feedrate scheduling method based on particle swarm optimization for a 5-DOF hybrid robot. *Robot. Comput.-Integr. Manuf.* **2022**, *78*, 102396. [CrossRef]
32. Fu, L.; Lei, Y.; Zhu, L.; Lv, J. An evaluation and design method for Ming-style furniture integrating Kansei engineering with particle swarm optimization-support vector regression. *Adv. Eng. Inform.* **2024**, *62*, 102822. [CrossRef]
33. Holland, J. *Adaptation in Natural and Artificial Systems*; University of Michigan Press: Ann Arbor, Michigan, 1975.
34. Xiang, R.; Feng, B. Path Optimization of Two-Posture Manipulator of Apple Packing Robots. *Appl. Sci.* **2024**, *14*, 8849. [CrossRef]
35. Zhao, Y.; Shuai, L.; Zhang, H.; Zheng, Y. Angle Control Algorithm for Air Curtain Based on GA Optimized Quadratic BP Neural Network. *Buildings* **2024**, *14*, 3144. [CrossRef]
36. Tegegne, G.; Kim, Y.-O. Representing inflow uncertainty for the development of monthly reservoir operations using genetic algorithms. *J. Hydrol.* **2020**, *586*, 124876. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.