

ORIGINAL RESEARCH

Open access

Asynchronous Weighted Multi-Fan Control for Energy-Efficient Server Thermal Management

Burak Demir^{1*}, Elif Arslan², Mehmet Kaya¹, Selin Kurt²

Abstract

Each server in data centers contains multiple fans and various heat-generating components. Factors such as fan speeds, component layout, and workload conditions affect airflow and temperature distribution within the server. Consequently, server thermal management is regarded as a Multi-Input Multi-Output (MIMO) nonlinear control system that cannot be effectively analyzed or controlled using conventional linear system techniques. This study implements a multi-fan thermal control approach for servers by applying a weighting mechanism that assigns distinct weights to each cooling fan, enabling asynchronous adjustment of multiple fans. This is combined with the Evolutionary Strategy (ES) algorithm and carefully designed fitness functions. The proposed system can identify near-optimal weighting combinations within a small number of search iterations, tailored to various workload scenarios. By asynchronously adjusting the fans according to these weightings, the system achieves near-optimal energy efficiency while still satisfying the server's thermal requirements—specifically, the maximum allowable temperatures for the CPU and PCIe components. Compared with traditional synchronous fan modulation, this asynchronous multi-fan control method reduces the total fan power consumption by an average of 43.1%. It requires searching only 14.2% of all possible weighting combinations across the eight experimental workload scenarios. Moreover, the probability that each identified near-optimal weighting corresponds to the global optimal solution is 47.5%. Experimental results confirm that the proposed asynchronous multi-fan control system successfully meets thermal specifications while delivering near-optimal energy savings for the server. Therefore, the developed approach proves practical and highly effective for substantial energy reduction, without requiring the development of mathematical thermal models or the analysis of extensive datasets. In the future, by fine-tuning the system's code parameters, it could potentially be adapted to different server architectures.

Keywords Multi-fan control system with asynchronous fan weightings, Server thermal control, Energy-saving optimization, Evolutionary strategy

*Correspondence:

Burak Demir
burak.demir@metu.edu.tr

¹ Department of Smart Grid Technologies, Faculty of Engineering, Middle East Technical University, Ankara, Turkey

² Department of Energy Analytics, Faculty of Electrical Engineering, Istanbul Technical University, Istanbul, Turkey

Introduction

The swift advancement of 5G infrastructure, Artificial Intelligence (AI), and Internet of Things (IoT) technologies—further accelerated by the global disruptions caused by the COVID-19 pandemic—has dramatically accelerated the shift toward digital operations across industries worldwide. Leading cloud providers, notably Amazon, Microsoft, and Google, have reaped substantial benefits from this trend [1] and continue to record reliable revenue expansion in the international marketplace. Their ongoing heavy capital commitments to infrastructure have driven the rapid proliferation of very large-scale data centers; estimates indicate that more than 1,000 hyperscale facilities will be in operation worldwide by 2024

[2]. Already in 2016, worldwide data centers were consuming over 300 terawatt-hours (TWh) of electricity annually [3]—an amount comparable to the yearly power demand of a typical developed country. Looking ahead, projections indicate that global data center electricity usage could reach 1,700 TWh per year by 2030 [4, 5], accounting for roughly 2.13% of the planet's total electricity consumption [6].

Today, cloud service operators are strongly committed to adopting sustainable, low-carbon energy practices in their data center operations. These initiatives serve dual purposes: fulfilling international climate obligations, such as those outlined in the Paris Agreement [7] and the Glasgow Climate Pact [8], while simultaneously reducing electricity costs and improving overall energy efficiency. The most widely adopted benchmark for measuring data center energy performance is Power Usage Effectiveness (PUE) [9], which is computed by dividing the facility's total power draw by the portion consumed exclusively by information technology hardware. A perfect score of 1.0 would mean that every watt entering the facility powers only IT equipment. In actual facilities, however, a PUE of exactly 1 is unattainable because electricity also supports auxiliary systems including transformers, backup power units, interior lighting, and—most significantly—cooling infrastructure [10]. Cooling systems account for approximately 37% of total power consumption (second only to IT equipment), and within that category, fans for air movement consume about 12% of the facility's electricity. Modern data centers may contain tens to tens of thousands of individual servers, each representing the smallest independent compute node. Cost considerations related to both initial build-out and long-term operation have kept traditional air-cooling as the prevailing method for removing heat from servers, thereby creating substantial electricity demand from fan operation. For this reason, the development of intelligent fan-speed regulation strategies that reliably maintain thermal safety limits while minimizing energy waste has become one of the most important research directions in modern data center design.

Most air-cooled servers are fitted with 4 to 8 dedicated cooling fans that extract heat from temperature-sensitive components such as the Central Processing Unit (CPU), Graphics Processing Unit (GPU), and devices connected via Peripheral Component Interconnect Express (PCIe). Within any given server, numerous heat-generating elements interact, and the resulting thermal behavior depends on hardware characteristics, component placement geometry, and current workload patterns. Close physical proximity among heat sources often leads to thermal coupling, while changes in individual fan speeds reshape internal airflow. These interdependencies greatly complicate thermal regulation and system-level modeling. Accordingly, thermal management in servers is widely recognized as a nonlinear Multi-Input Multi-Output (MIMO) control challenge [11].

Shin et al. [12] described what is now commonly called the fan law, which mathematically links fan power consumption to the cube of rotational speed and shows a linear relationship between fan speed and duty cycle. Intelligent, demand-based fan speed adjustment—avoiding prolonged operation at unnecessarily high duty cycles—offers one of the most effective ways to eliminate wasteful power usage.

Wang et al. [11] conducted detailed measurements on a multi-blade server rack equipped with shared fans. Drawing on observed workload levels, processor core temperatures, incoming air temperatures, and manufacturer fan curves, the authors derived a thermal resistance expression, fitted model coefficients via regression techniques, and constructed a comprehensive thermal transfer model of the system. They also employed heat maps to reveal the complex mutual thermal interactions among neighboring servers and fans. Taking advantage of the cubic fan power–speed relationship, the team developed a MIMO-based zoned fan controller to improve overall thermal regulation performance.

Lee and Chen [13] presented an adaptive PID control scheme that simultaneously tunes fan speeds. Bypassing the need for an explicit system model, their method leveraged neural networks to continuously refine controller gains in real time, balancing thermal safety constraints and energy conservation priorities.

Acun [14] assembled a comprehensive dataset capturing CPU power draw, core usage rates, clock frequencies, temperatures at various points, fan speeds, and surrounding air temperature across a wide range of workloads. After appropriate preprocessing, a neural network-based temperature predictor was trained using the Levenberg-Marquardt backpropagation algorithm. The resulting predictive model was integrated with forward-looking (preemptive) controllers

that evaluated multiple candidate fan speed combinations, ultimately selecting the most favorable combination according to a defined cost function, thereby enabling anticipatory fan speed management.

More recently, data-driven techniques and digital-twin concepts have gained traction in server thermal management research. Brännvall et al. [15] demonstrated a digital-twin-assisted calibration procedure for fan control logic that markedly improved cooling effectiveness. Sarkinen et al. [16] experimentally evaluated several strategies for independently controlling fans within individual servers, offering practical lessons that advance thermal management practices in real-world data centers. Together, these contributions provide important context and complementary ideas for the present study.

Chu et al. [17] adopted several simplifying assumptions about server behavior and constructed an empirical thermal model [16]. They then collected real measurement data from heat-generating components. They applied reinforcement learning—specifically the Deep Deterministic Policy Gradient (DDPG) method—while carefully designing reward signals to train an energy-aware fan control policy.

The same group (Chu et al. [18]) proposed categorizing the multiple parallel fans inside a server into primary and secondary groups. By systematically testing diverse workload patterns together with various fan duty-cycle assignments, they investigated thermal decoupling behavior. Subsequently, they trained two advanced deep learning architectures—MTSMFF (Multivariate Time Series Forecasting Framework) and MSA (Multistage Attention)—to perform accurate multi-step temperature forecasting.

Several earlier works [13, 14, 17] adjust all server fans in unison (synchronous control). Research reported in Lee and Chen [13] and Acun [14] argues that assigning individualized weightings to fans [19] and modulating them asynchronously could lessen excessive cooling of certain components (a common drawback of uniform adjustment) and thereby further decrease fan-related power draw. Unfortunately, those publications offer little detail on how to determine or choose such weight values. Moreover, the machine-learning models described in Acun [14], Chu et al. [17], and Chu et al. [18] typically require very large training and validation datasets, resulting in high computational costs during both the development and inference phases. Long-term maintainability of such models also warrants careful consideration, since swapping out hardware components can modify heat generation and dissipation patterns, while changing to a different fan model can alter airflow and cooling capability. In deployed systems, fan control logic is ordinarily executed by the Baseboard Management Controller (BMC) chip. Because the models proposed in Acun [14], Chu et al. [17], and Chu et al. [18] impose considerable processing burdens, substantial optimization efforts will be required before they can be practically hosted on resource-constrained BMC hardware.

Based on the hardware platform described in Chu et al. [18], the present work introduces an innovative multi-fan control approach. Under this approach, each fan is adjusted independently through a dedicated weighting mechanism. This weighting-based multi-fan control is then combined with the Evolution Strategy (ES) algorithm to form a complete server multi-fan regulation system capable of maintaining thermal constraints while optimizing for energy efficiency. The proposed framework eliminates the need to develop a detailed heat transfer model and enables online parameter tuning without large-scale data collection, thereby achieving notably lower implementation costs. In real-world applications, multi-fan control is realized simply by configuring the appropriate fan weight values—matched to specific server workload scenarios—directly within the Baseboard Management Controller (BMC) chip.

This investigation exclusively employs the server hardware outlined in Chu et al. [18]. It builds upon the fan weighting principles presented in Lee and Chen [13], Acun [14], and Byquist and Crocker [20] by incorporating a weighting mechanism that enables asynchronous adjustment of the multiple fans. By coupling this mechanism with the Evolutionary Strategy algorithm and carefully crafting suitable fitness functions, the study develops a multi-fan control solution for the server that satisfies thermal requirements and delivers near-optimal energy savings. No explicit heat transfer model is required, and because fan weightings are optimized in real time, the method avoids the need to collect extensive training

datasets, keeping deployment costs minimal. In practical deployment, entering the workload-specific fan weighting values into the server's BMC chip is sufficient to implement differentiated, asynchronous multi-fan control.

Experimental System

All experiments in this research were performed using a 2U rack-mount server, as shown in **Figures 1-3**. The server measures 450 mm (width) × 760 mm (depth) × 90 mm (height) and is built around an Inventec N880G3 DC motherboard. It is equipped with two Intel Xeon E5-2699 v3 processors, each rated at 144 W. The system also includes twelve TOSHIBA MG04ACA500E hard disk drives, sixteen SK Hynix DDR4-2666 16GB 2Rx4 PC4-2133P memory modules, and two custom-designed PCIe cards (each with a peak power draw of 24 W). Power is supplied by two 750 W power supply units (PSUs). Cooling is provided by four Delta Electronics PFR0812UHE fans, each measuring 80 mm (width) × 80 mm (depth) × 38 mm (height) and capable of reaching a maximum speed of 8000 RPM. These four-pin fans include a ground line (black wire), a 12 V power input (red wire), a tachometer output for speed monitoring (yellow wire), and a PWM control input (blue wire), as illustrated in **Figure 4**.

Following the fan law principles outlined in Shin et al. [12], individual fan characterization tests were carried out to measure power consumption across a range of duty cycles. The resulting data, presented in **Figure 5** and captured by **Eq. 1**, were fitted to a cubic polynomial relating fan power (P) to duty cycle (R), with the remaining symbols representing fitted coefficients. This cubic model was then applied to compute the combined power draw of all four fans by summing their individual calculated power values.

$$P(R) = (8.418e - 7) R^3 + (0.001761) R^2 + (0.08436) R + 3.249 \quad (1)$$



Figure 1. 2U server.



Figure 2. The real location map of the internal components of the server.

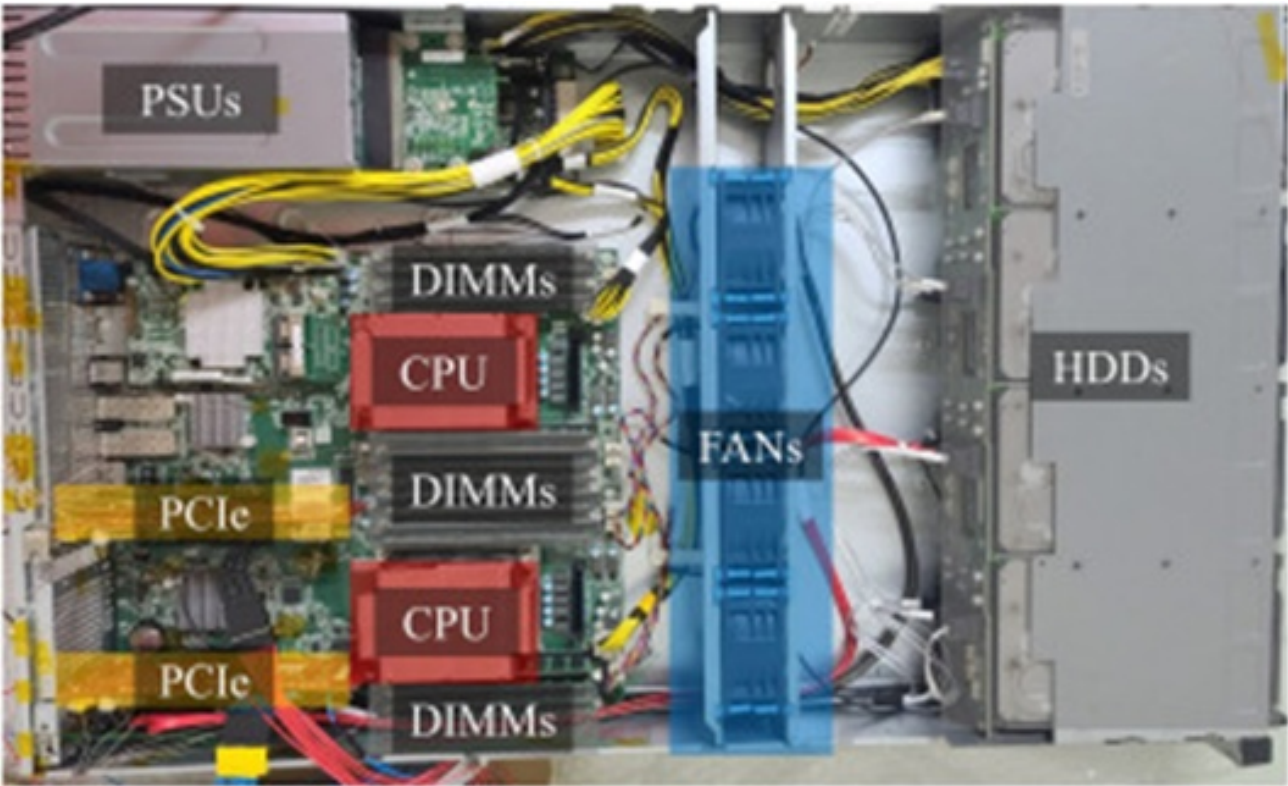


Figure 3. Schematic diagram of the internal components of the server.



Figure 4. Delta Electronics PFR0812UHE server fan.

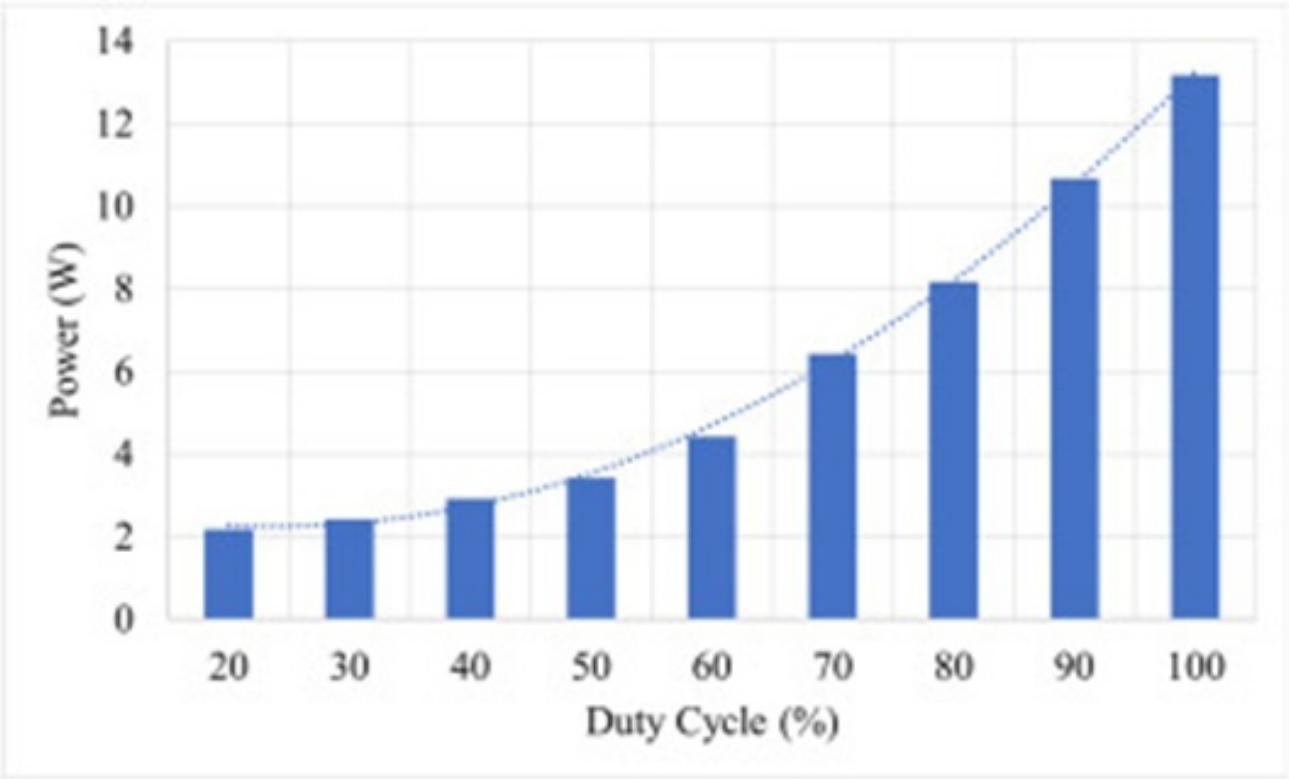


Figure 5. Power variation at different fan duty cycles.

The layout of the server’s internal components is shown in [Figure 3](#). During the experiments, a wind scooper was mounted immediately above each CPU to channel airflow more effectively toward the heat sinks, thereby improving thermal performance. The server chassis cover remained installed throughout testing. The placement and configuration of the wind scoopers are shown in [Figure 6](#).



Figure 6. Placement of wind scooper.

The server is equipped with Power Thermal Utility (PTU) software, which enables dynamic adjustment of CPU power levels and incorporates the Intelligent Platform Management Interface (IPMI) to facilitate remote oversight of server operations. In the experiments described here, PTU was utilized to vary CPU power consumption, thereby generating a range of different workload conditions for testing. Due to hardware constraints, however, neither temperature monitoring of server components nor fan speed regulation could be performed directly via the Baseboard Management Controller (BMC) chip. Instead, the setup relied on K-type thermocouples, an Agilent 34970A data acquisition unit, and an external PWM signal generator. An Arduino Mega 2560 board was employed to operate a relay module that controlled power delivery to the PCIe cards. Since the CPUs and PCIe cards are the primary sources of heat, thermocouples were strategically positioned at their hottest cores to capture accurate temperature readings. These measured temperatures served as feedback signals for the control system developed in this study. In response, the control system generated appropriate duty-cycle instructions, which were transmitted to the PWM signal generator and subsequently applied to the fans' PWM input pins, thereby implementing active fan speed regulation for the server. This overall experimental architecture is illustrated in **Figure 7**.

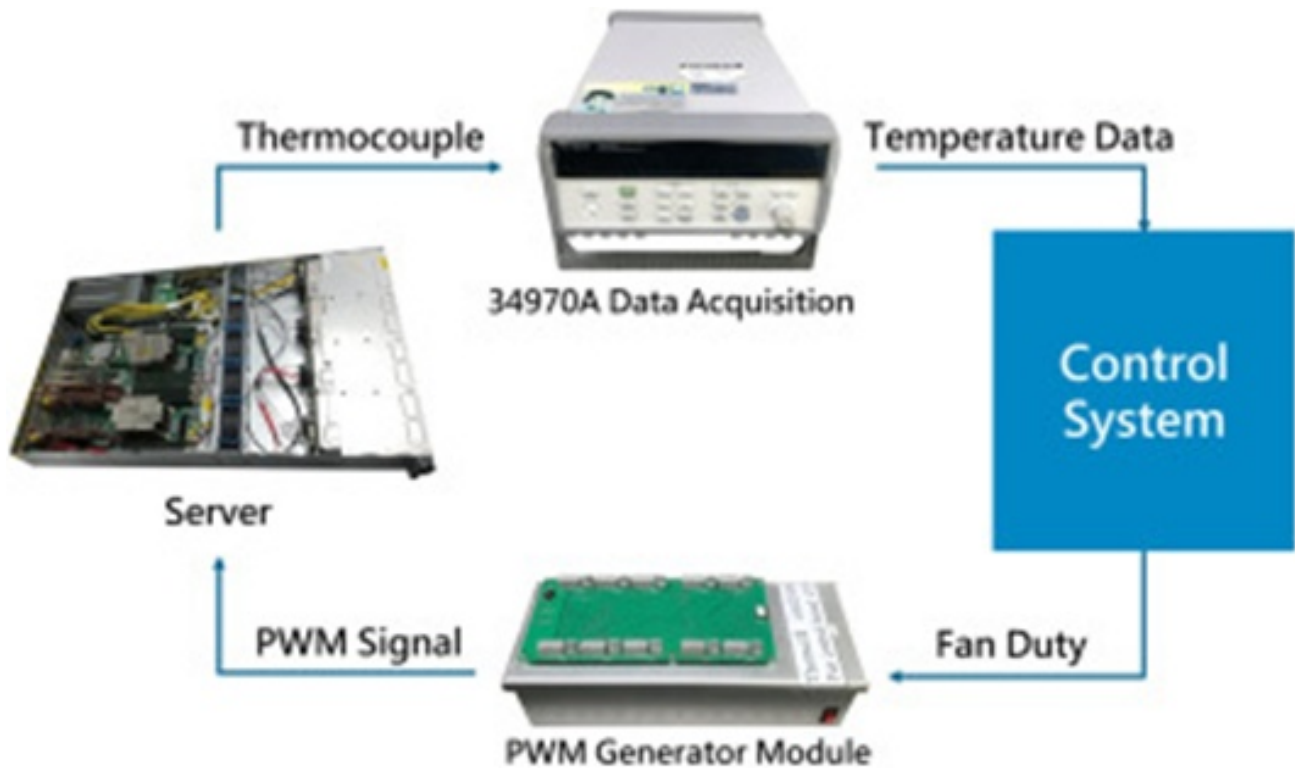


Figure 7. Schematic of experimental system.

The weighting mechanism introduced in this work draws directly on the concepts described by Lee and Chen [13], Acun [14], and Byquist and Crocker [20]. By integrating a weighting matrix into the controllers, the system enables independent, asynchronous adjustment of the multiple fans, as shown in Figure 8.

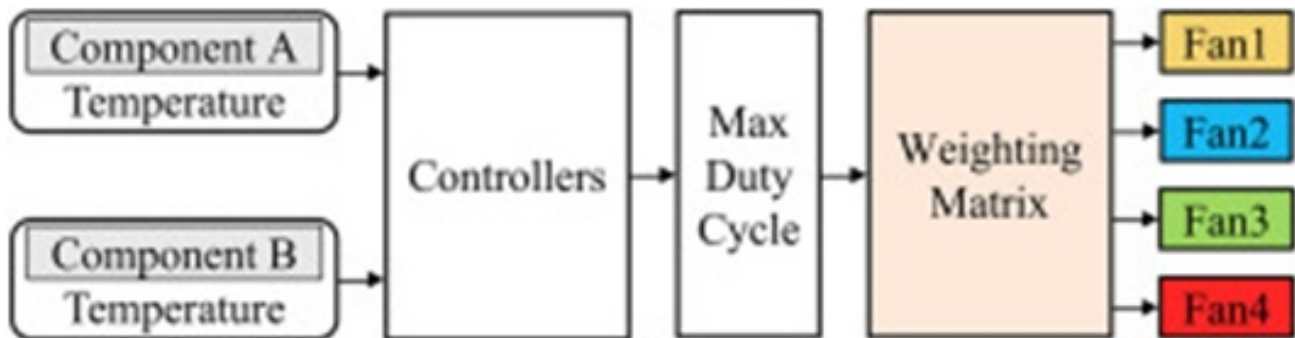


Figure 8. Asynchronous fans modulation.

In response to observed temperature variations, the controllers continuously recalculate and update the maximum duty cycle. The individual elements of the resulting matrix then define the specific duty cycle applied to each corresponding fan, as expressed in Eq. 2. Here, Weighting fan denotes the weighting factor assigned to each fan, with values ranging from 0.0 to 1.0. Setting all fan weightings to 1.0 results in uniform (synchronous) fan modulation rather than differentiated (asynchronous) control. Max Duty Cycle is the maximum duty cycle output by the controllers at any given moment. *Fan* indicates the actual duty cycle command sent to the respective fan. Although the choice of controller type is flexible, this study adopts the PID (Proportional-Integral-Derivative) controller [18] owing to its clear physical interpretability of parameters, straightforward implementation, and widespread adoption as the industry-standard controller. The PID controller achieves closed-loop regulation by utilizing feedback from the controlled process. Eqs give its transfer function

$G_c(s)$. 3, while Eqs describe the time-domain control output $u(t)$. 4. K_p is the proportional gain, which provides immediate corrective action proportional to the current error. K_i is the integral gain, which accumulates past errors to eliminate the steady-state offset. K_d is the derivative gain, which anticipates future error trends by responding to the rate of change of the error, thereby enhancing settling time and overall system stability. s denotes the complex frequency-domain variable in the Laplace transform. $e(t)$ is the error signal, calculated as the difference between the desired setpoint and the actual measured value of the controlled variable. t represents elapsed time. Users need only tune the three parameters K_p , K_i , and K_d to achieve the targeted control performance. Given these practical benefits, the PID controller was selected for this research. It continually adjusts the maximum duty cycle output in response to real-time temperature feedback from the key heat-generating components. By appropriately tuning K_p , K_i , and K_d , users can ensure the system response meets the required design criteria. Leveraging these strengths, this study implements PID-based feedback control to regulate the temperatures of the most critical components.

$$\begin{bmatrix} \text{Weighting fan1} \\ \text{Weighting fan2} \\ \text{Weighting fan3} \\ \text{Weighting fan4} \end{bmatrix} \times [\text{Max Duty Cycle}] = \begin{bmatrix} \text{Fan1} \\ \text{Fan2} \\ \text{Fan3} \\ \text{Fan4} \end{bmatrix} \quad (2)$$

$$G_c(s) = K_p + \frac{K_i}{s} + K_d s \quad (3)$$

Evolutionary Strategy Algorithm

Evolutionary Algorithms (EAs) [21] constitute a family of metaheuristic optimization techniques within artificial intelligence. This class encompasses methods such as the genetic algorithm (GA), evolution strategy (ES), and genetic programming (GP), among others. Drawing inspiration from Darwinian principles of natural selection and evolution [22], EAs simulate biological evolutionary processes to refine populations of candidate solutions toward superior outcomes progressively. These algorithms are versatile and effective across diverse optimization tasks, offering advantages such as shorter computation times and a reduced risk of converging to suboptimal local minima compared with conventional gradient-based numerical approaches.

The evolution strategy (ES) is specifically designed for problems involving multidimensional real-valued parameters, directly representing solutions as real-number vectors that evolve over generations. It depends primarily on mutation as the main variation operator, with mutation parameters that self-adapt throughout the process, eliminating the need for manual adjustment. In addition, ES typically converges faster than GA or GP, making it particularly popular for engineering parameter optimization tasks. A flowchart outlining the ES procedure is shown in **Figure 9**. The algorithm starts by initializing a population, then generates offspring until a population of size λ is obtained. Fitness values are evaluated for every individual in the current population. The μ best-performing individuals (according to fitness) are selected to serve as parents for the next generation. Offspring are subsequently created through recombination followed by mutation. This cycle—excluding the initial setup—repeats iteratively, refining the population and improving fitness scores until a predefined stopping criterion is satisfied. Such criteria may include reaching a maximum number of generations, observing sufficient population diversity, attaining a desired fitness threshold, or detecting convergence. Typical parameter configurations for ES are summarized in **Table 1**, where each individual is encoded as a real-valued vector. Discrete recombination involves randomly choosing parent vectors and directly inheriting them. Mutation introduces stochastic perturbations by adding Gaussian noise to each dimension of the vector [23]. The Gaussian distribution $N(\xi, \sigma)$ is characterized by two parameters: ξ (the mean, conventionally set to 0) and σ (the standard

deviation, which controls the scale of variation and is referred to as the mutation step size—a pivotal tunable element in the algorithm).

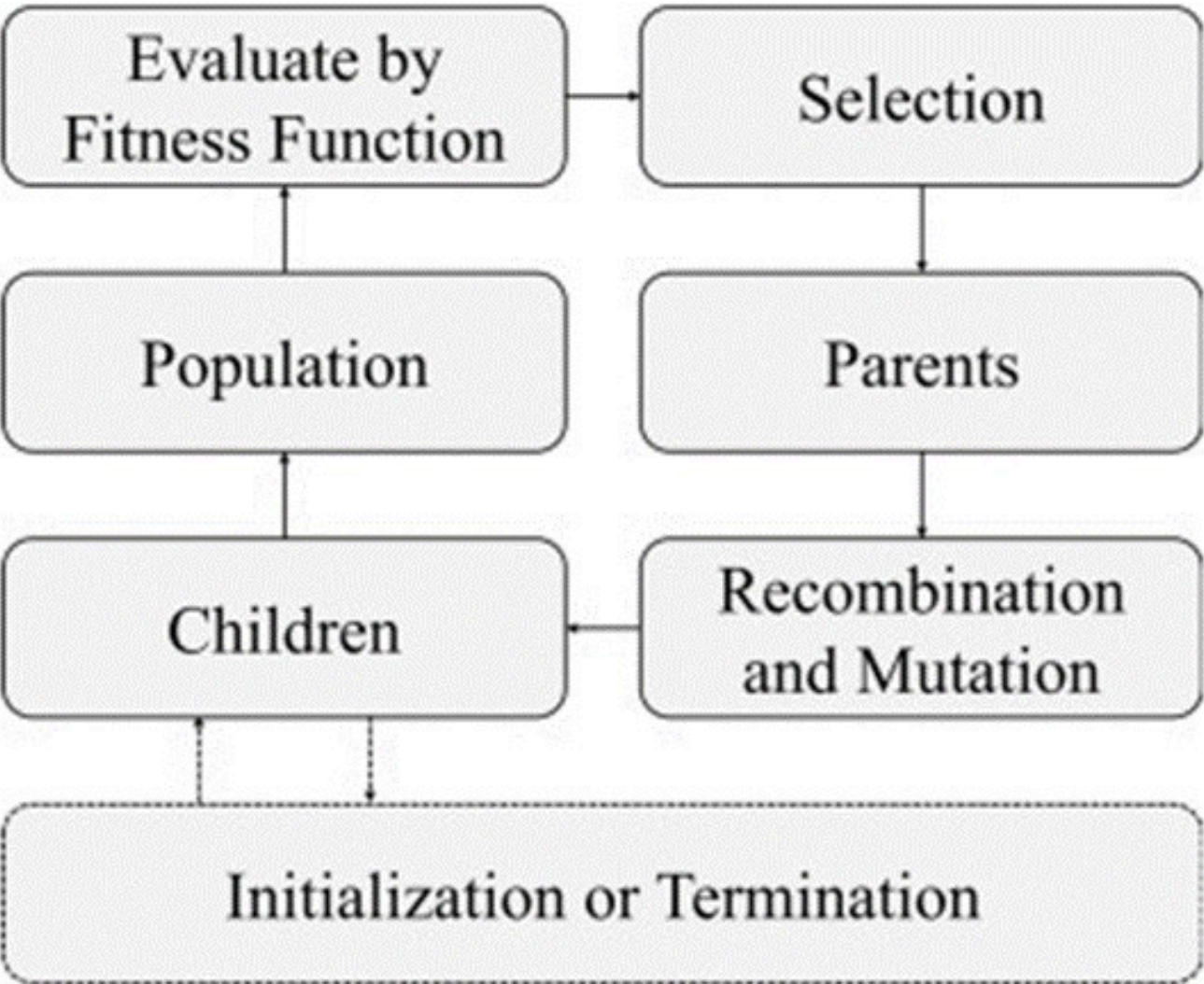


Figure 9. The flowchart of the ES algorithm.

Table 1. Common parameter settings for the evolution strategy.

Component	Description
Encoding scheme	Continuous real-number vectors
Crossover method	Discrete recombination
Mutation operator	Gaussian-based perturbation
Parent selection strategy	Uniform random selection
Survivor replacement strategy	(μ , λ) selection scheme
Distinctive feature	Self-adaptive mutation step-size adjustment

The mutation step size can be grouped into two main types: parameter tuning and parameter control. In parameter tuning, the mutation step size stays fixed throughout the entire evolutionary process. In contrast, parameter control allows the mutation step size to vary dynamically as evolution proceeds. Parameter control itself can be subdivided into deterministic, adaptive, and self-adaptive approaches. In general, self-adaptive parameter control is preferable, as it allows the mutation step size σ to evolve with each iteration. When solution quality is low, σ increases to promote greater diversity and explore a wider search space. When solution quality is high, σ decreases to limit variation and concentrate the search in promising nearby regions.

In addition to the self-adaptive approach, three distinct variants exist [21, 24]:

- Uncorrelated mutation with one step-size, in which all dimensions of the vector share a single common mutation step size (Figure 10),
- Uncorrelated mutation with n step-sizes, where each dimension of the vector is assigned its own independent mutation step size $\sigma_1, \sigma_2, \dots, \sigma_n$, and each evolves separately (Figure 11),
- Correlated mutation, where each dimension has its own mutation step size $\sigma_1, \sigma_2, \dots, \sigma_n$, supplemented by rotation angles $\alpha_1, \alpha_2, \dots, \alpha_{n(n-1)/2}$ defined between every pair of dimensions, with both the step sizes and rotation angles adapting over iterations (Figure 12).

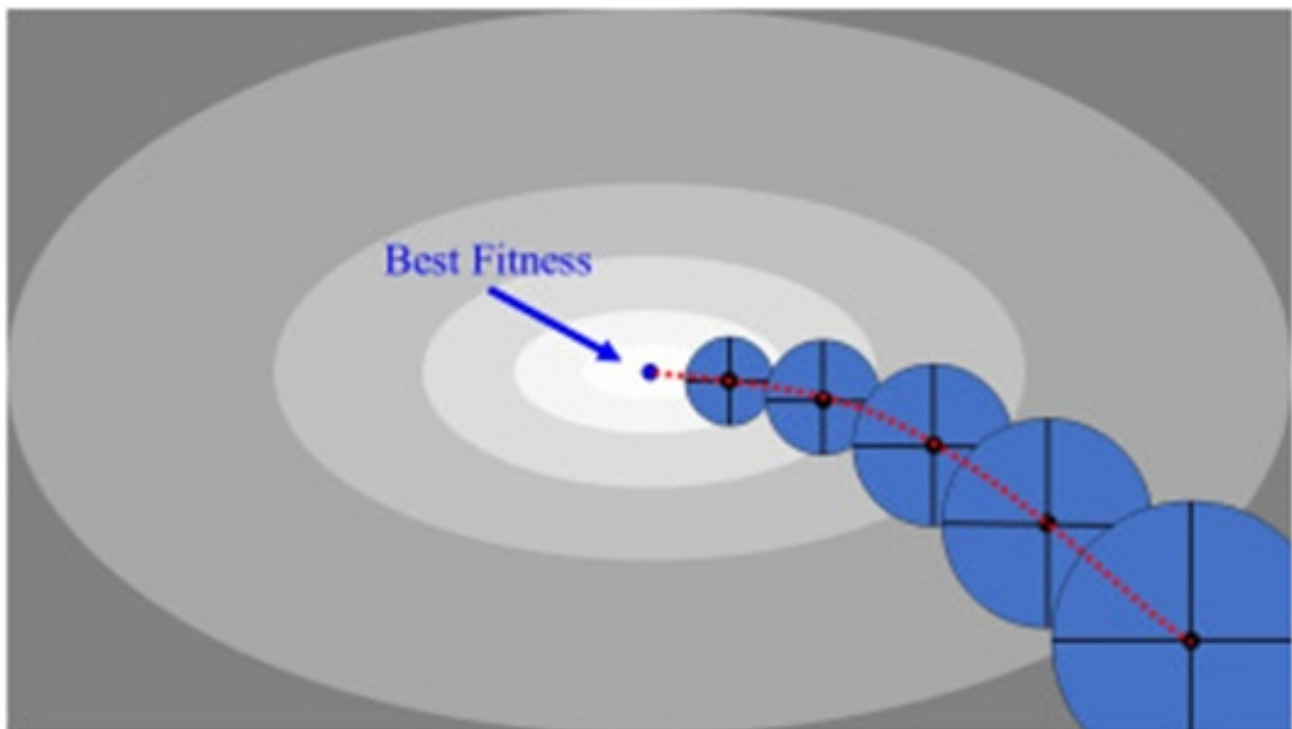


Figure 10. Uncorrelated mutation with one step-size.

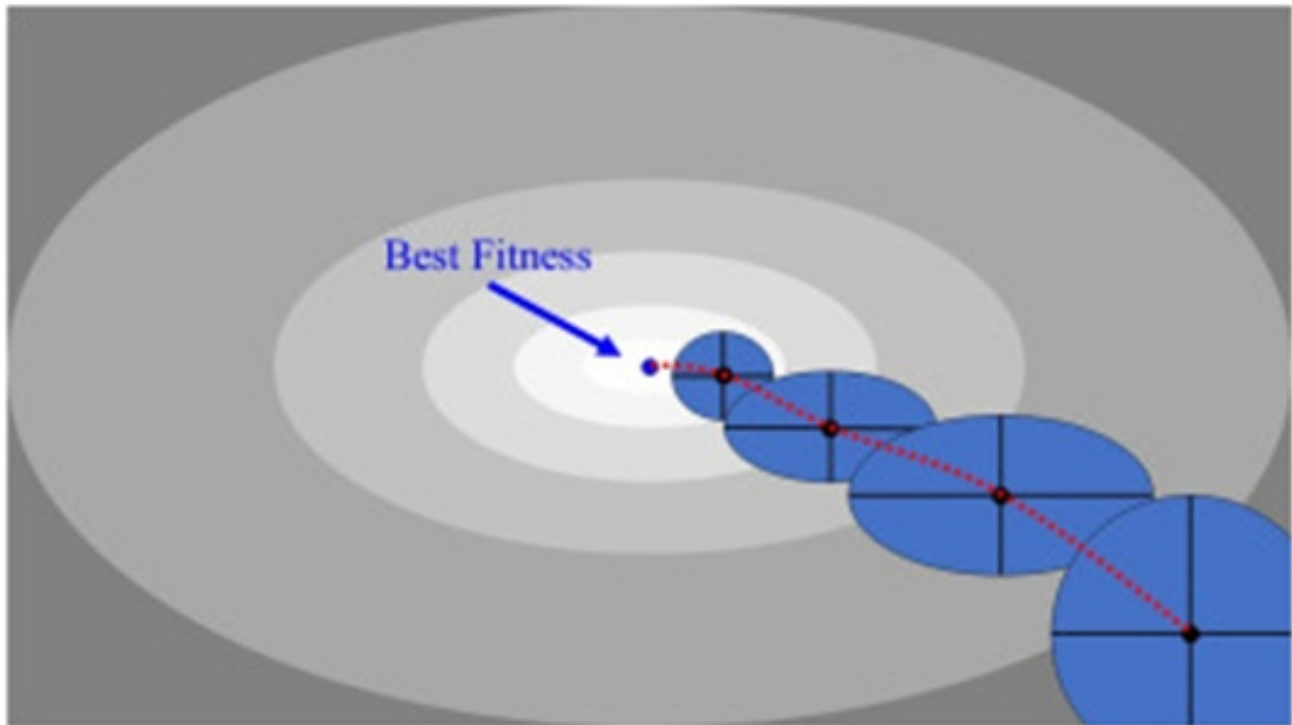


Figure 11. Uncorrelated mutation with n step-sizes.

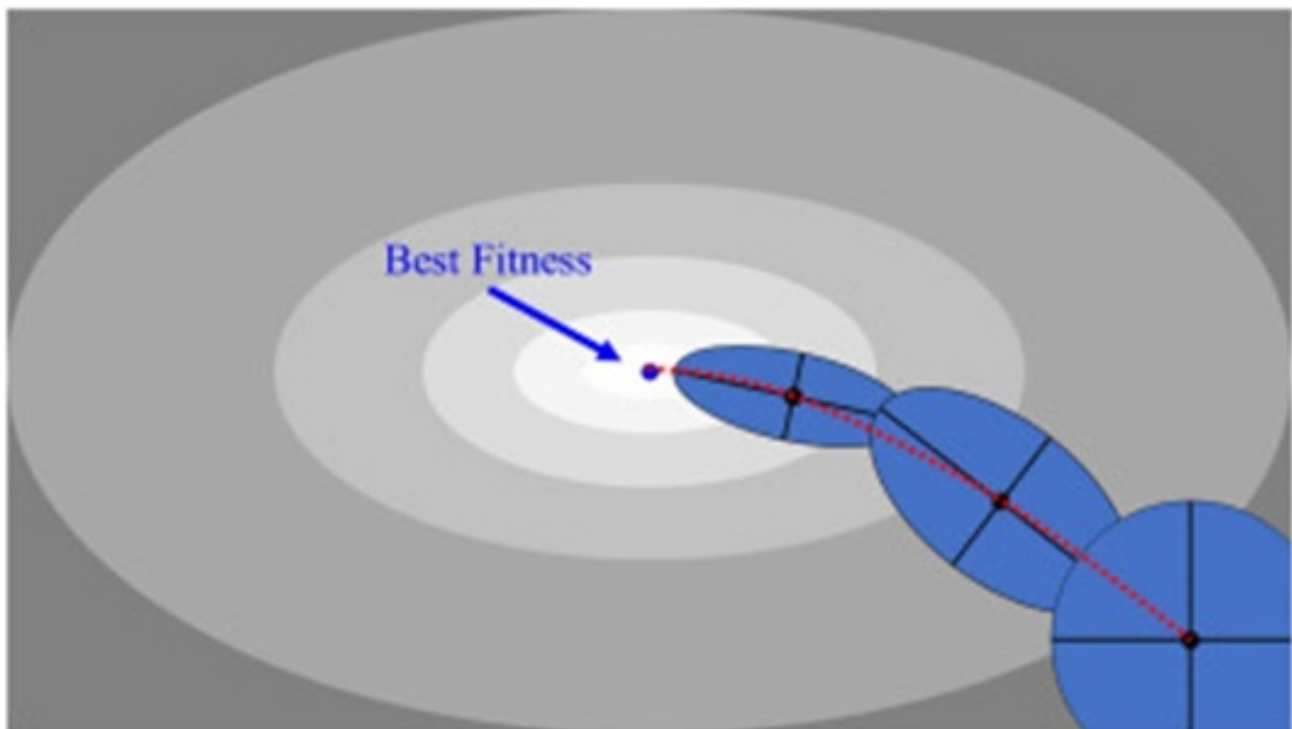


Figure 12. Correlated mutation.

Among the self-adaptive variants, Correlated Mutation typically achieves the strongest performance in locating optimal solutions. Nevertheless, it requires adapting a large number of parameters (a real-valued vector, a mutation step size per dimension, and rotation angles for each pair of dimensions), which often results in slow convergence. Hansen and

Ostermeier [25] introduced an improved variant of Evolution Strategy known as the Covariance Matrix Adaptation Evolution Strategy (CMA-ES). CMA-ES employs the mean of selected parents, the covariance matrix, a global mutation step size, and evolution paths as its adaptive parameters, thereby replacing the numerous parameters used in correlated mutation. This modification overcomes the limitation on convergence speed and enables correlated mutation to be applied more effectively. Since its introduction, CMA-ES has gained widespread adoption, with subsequent refinements leading to the most recent version released in 2016 [26].

Furthermore, real-world engineering optimization tasks are frequently complex—for instance, multi-funnel functions [27] that feature multiple local optima and lack any inherent concave structure that guides them toward a single global optimum. Consequently, most practical engineering applications prioritize the discovery of near-optimal (approximately optimal) solutions rather than insisting on true global optima.

To quickly determine fan weightings that satisfy temperature constraints while approaching energy-saving objectives, this study implements a multi-fan control system. The system uses a weighting mechanism to adjust fans asynchronously and applies CMA-ES to identify near-optimal fan weightings, thereby achieving effective thermal management alongside energy-saving optimization.

Two fitness functions were formulated to assess the quality of fan weightings according to the outcomes of the multi-fan control process. If a given fan weighting causes the multi-fan control outcome to violate the temperature specifications, its fitness is computed using Eq. 4. The greater the deviation between the ideal time and the actual time under fan control, the more rapidly that weighting breaches the temperature limits—the final term in Eq. 4, “100% Fans Power,” serves as a strong penalty, informing the algorithm that this weighting is unsuitable for the multi-fan control system because of the temperature violation. On the other hand, if the fan weighting complies with all temperature specifications, its fitness is determined by Eq. 5. For such valid weightings, the fitness value corresponds directly to the total fan power observed in the control outcome, with total fan power calculated via Eq. 1. Lower fitness values indicate higher-quality fan weightings.

$$f(x_k) = (IdealTime - ActualTime) + 100 \text{ Fans Power} \quad (4)$$

$$f(x_k) = FansPower \quad (5)$$

In Algorithm 1 and 2, Step 1 handles parameter initialization. Since the fan weighting comprises Weighting fan1, fan2, fan3, and fan4, corresponding to a four-dimensional optimization problem, the dimension n is set to 4. The number of offspring per generation, λ , is set to 8. The number of selected parents, μ , is set to 4, with the initial individual-weighted values defined as $(w_1, w_2, w_3, w_4) = (0.53, 0.29, 0.14, 0.04)$, as given in Eqs. 6 and 7. Here, w_i represents the weighted value associated with the fitness rank of the i th individual among the μ selected parents.

n denotes the problem dimension. f denotes the fitness function used to evaluate offspring. x_k denotes the k th offspring, consisting of fan weights. Parameters including the initial global mutation step size σ^0 , initial covariance matrix $C^{(0)}$, initial parents mean $M^{(0)}$, initial covariance matrix evolution path $P^{(0)}$, initial global step-size evolution path $P^{(0)}$, and related constants μ_{eff} , c_1 , c_c , c_μ , c_σ , and d_σ adopt the default values recommended by the algorithm's authors. As noted in Schwefel [24] and Hansen [26], these defaults are robust and suitable for a wide range of function optimization problems. In Algorithm 2, Step 2 checks the termination criterion. Step 3 performs recombination and mutation using the parents' mean $M^{(g)}$, global mutation step size $\sigma^{(g)}$, and $N(0, C^{(g)})$ to generate λ offspring, where g denotes the g th generation.

Here, $N(0, C^{(g)})$ represents a multivariate Gaussian distribution with zero mean and covariance matrix $C^{(g)}$. Step 4 evaluates the fitness of each offspring using the fitness function. Steps 5–10 update key parameters, including the new parents' mean $M^{(g+1)}$, the covariance matrix evolution path $P^{(g+1)}$, the covariance matrix $C^{(g+1)}$, and others. In particular, $x^{(g+1)}$ refers to the i th best offspring selected from the λ offspring in generation $g+1$, where $i:\lambda$ indicates ranking among the λ offspring. The weights w_i are used during selection to form the parent population. Steps 3–10 repeat iteratively until the termination condition is met, after which the evolutionary search concludes in Step 11.

$$w_i = \frac{\ln(n+0.5) - \ln(i)}{\sum_{i=1}^{\mu} \ln(n+0.5) - \ln(i)} \text{ for } i = 1, \dots, \mu \quad (6)$$

$$\sum_{i=1}^{\mu} w_i = 1 \quad (7)$$

Experimental Setup

Experiments were performed using constant server workload conditions. The surrounding air temperature was held steady at 23 °C throughout the test environment. The primary heat-producing elements—namely CPU0, CPU1, PCIe0, and PCIe1—were configured to run at either 60% or 100% of their full rated power, generating a total of eight unique workload profiles, as illustrated in [Figure 13](#). Thermal limits were defined such that the target temperature (setpoint) for each of these critical components was fixed at 55 °C. While fans were actively controlled, the transient temperature was not permitted to exceed 5 °C above this setpoint to avoid thermal damage. In the steady-state phase, temperature was allowed to exceed the setpoint by no more than 2%, establishing the maximum tolerable steady-state deviation. The layout of these heat-critical parts and the associated cooling fans within the server chassis is shown in [Figure 14](#).

For the weighting approach, individual Weightingfan values were restricted to the range 0.2–1.0, with a step size of 0.2. A key requirement was that at least one fan's weighting had to equal 1.0. This rule produced exactly 369 valid fan weighting combinations. All combinations are displayed in [Figure 15](#) and represent the complete set of possibilities explored by the weighting-based asynchronous fan control strategy in this study. An overview of the asynchronous multi-fan modulation experimental procedure is provided in [Figure 16](#).

PID controller gains were tuned to $K_p = 15$, $K_i = 0.5$, $K_d = 0.001$. Each experimental trial targeted a total fan control duration of 400 seconds. Synchronous fan operation was enforced during the first 200 seconds (0–200 s). From 200 to 400 seconds, asynchronous modulation was applied using different weighting sets. Temperature behavior during the transient phase was monitored from 200 to 300 seconds, while steady-state performance was evaluated over the final 100 seconds (300–400 s). Any weighting combination that caused a violation of the thermal constraints triggered immediate termination of that control trial, and the corresponding weighting was labeled inappropriate. Weightings that kept temperatures within limits throughout were deemed appropriate. Both synchronous and asynchronous control modes were compared with respect to total fan power usage and resulting component temperatures.

1 Initialize the CMA-ES Parameters

$n \leftarrow$ Number of dimensions

$\lambda \leftarrow$ Offspring Population size ($4 + \text{floor}(3 \ln(n))$)

$\mu \leftarrow$ Parent Population for next generation ($\text{floor}(\lambda/2)$)

$w_i \leftarrow$ Weight Value of i-th Individual

$f \leftarrow$ Function of Fitness

$\sigma^0 \leftarrow$ Initial Global Mutation Step Size

$\sigma^0 \leftarrow n \text{ dim. (uppest bound + lowest bound)}/6$

$\mathbf{C}^0 \leftarrow$ Initial Covariance Matrix is $\mathbf{I}_{n \times n}$

$\mathbf{M}^0 \leftarrow$ Initial Mean is n dim. (uppest bound + lowest bound)/2

$\mathbf{P}_c^0 \leftarrow$ Initial Cov. Matrix Evolution Path is $[0, \dots, 0]_{1 \times n}^T$

$\mathbf{P}_\sigma^0 \leftarrow$ Initial Mutation Step Size Evolution Path is $[0, \dots, 0]_{1 \times n}^T$

$\mu_{eff} \leftarrow$ Cov. Matrix & Mutation Step Size related parameter

$\mu_{eff} \leftarrow (\sum_i^\mu w_i^2)^{-1}$

$c_1, c_c, c_\mu \leftarrow$ Cov. Matrix related parameters

$c_1 \leftarrow 2((n + 1.3)^2 + \mu_{eff})^{-1}$

$c_c \leftarrow (4 + \mu_{eff}/n)(n + 4 + 2\mu_{eff}/n)^{-1}$

$c_\mu \leftarrow \min(1 - c_1, 2(\mu_{eff} - 2 + (\mu_{eff})^{-1})(n + 2)^2 + \mu_{eff})^{-1}$

$c_\sigma, d_\sigma \leftarrow$ Mutation Step Size related parameters

$c_\sigma \leftarrow (\mu_{eff} + 2)(n + \mu_{eff} + 5)^{-1}$

$d_\sigma \leftarrow 1 + 2 \max(0, \sqrt{(\mu_{eff} - 1)(n + 1)^{-1}} - 1)(n + \mu_{eff} + 5)^{-1}$

Algorithm 1. CMA-ES Pseudo Code - Part 1.

1 Initialize the CMA-ES Parameters (as Part1 show)

2 **while** stopping criterion is not met **do**

3 Generate Offspring Population for generation $g+1$

$\mathbf{x}_k^{(g+1)} \sim \mathbf{M}^{(g)} + \sigma^g \mathbf{N}(\mathbf{0}, \mathbf{C}^{(g)})$ for $k = 1, \dots, \lambda$

4 Evaluate Fitness

$f(\mathbf{x}_k^{(g+1)})$

5 Update Mean for generation $g+1$

$\mathbf{M}^{(g+1)} \leftarrow \sum_{i=1}^{\mu} w_i \mathbf{x}_{i:\lambda}^{(g+1)}$

6 Update Cov. Matrix Evolution Path

$\mathbf{P}_C^{(g+1)} \leftarrow$
 $(1 - c_c) \mathbf{P}_C^{(g)} + c_c (2 - c_c \mu_{eff})^{0.5} (\mathbf{M}^{(g+1)} - \mathbf{M}^{(g)}) (\sigma^g)^{-1}$

7 Update Cov. Matrix

$\mathbf{C}^{(g+1)} \leftarrow$
 $(1 - c_\mu) \mathbf{C}^{(g)} + c_\mu \mathbf{P}_C^{(g+1)} (\mathbf{P}_C^{(g+1)})^T$
 $+ c \sum_{i=1}^{\mu} w_i \mathbf{x}_{i:\lambda}^{(g+1)} w (\mathbf{x}^{(g+1)} - \mathbf{M}^{(g)}) (\sigma^g)^{-1} [(\mathbf{x}_{i:\lambda}^{(g+1)} - \mathbf{M}^{(g)}) (\sigma^g)^{-1}]^T$

8 Update Mutation Step Size Evolution Path

$\mathbf{P}_{\sigma}^{(g+1)} \leftarrow$
 $(1 - c_\sigma) \mathbf{P}_{\sigma}^{(g)}$
 $+ c_\sigma (2 - c_\sigma) \mu_{eff} (\mathbf{C}^{(g)})^{-1/2} (\mathbf{M}^{(g+1)} - \mathbf{M}^{(g)}) (\sigma^g)^{-1}$

9 Update Mutation Step Size

$\sigma^{g+1} \leftarrow \sigma^{g+1} \exp(c_\sigma d_\sigma^{-1} \|\mathbf{P}_{\sigma}^{(g+1)}\| (E[\|\mathbf{N}(\mathbf{0}, \mathbf{I})\|])^{-1} - 1)$

10 Update best ever fitness solution

11 **end**

Algorithm 2. CMA-ES Pseudo Code - Part 2.

Case A		Case B		Case C		Case D	
PCIe1	CPU1	PCIe1	CPU1	PCIe1	CPU1	PCIe1	CPU1
100%	100%	60%	100%	100%	100%	60%	100%
PCIe0	CPU0	PCIe0	CPU0	PCIe0	CPU0	PCIe0	CPU0
100%	100%	60%	100%	60%	60%	100%	60%

Case E		Case F		Case G		Case H	
PCIe1	CPU1	PCIe1	CPU1	PCIe1	CPU1	PCIe1	CPU1
60%	60%	100%	60%	60%	60%	100%	60%
PCIe0	CPU0	PCIe0	CPU0	PCIe0	CPU0	PCIe0	CPU0
60%	60%	100%	60%	100%	100%	60%	100%

Figure 13. Eight cases of the server's loading scenarios.

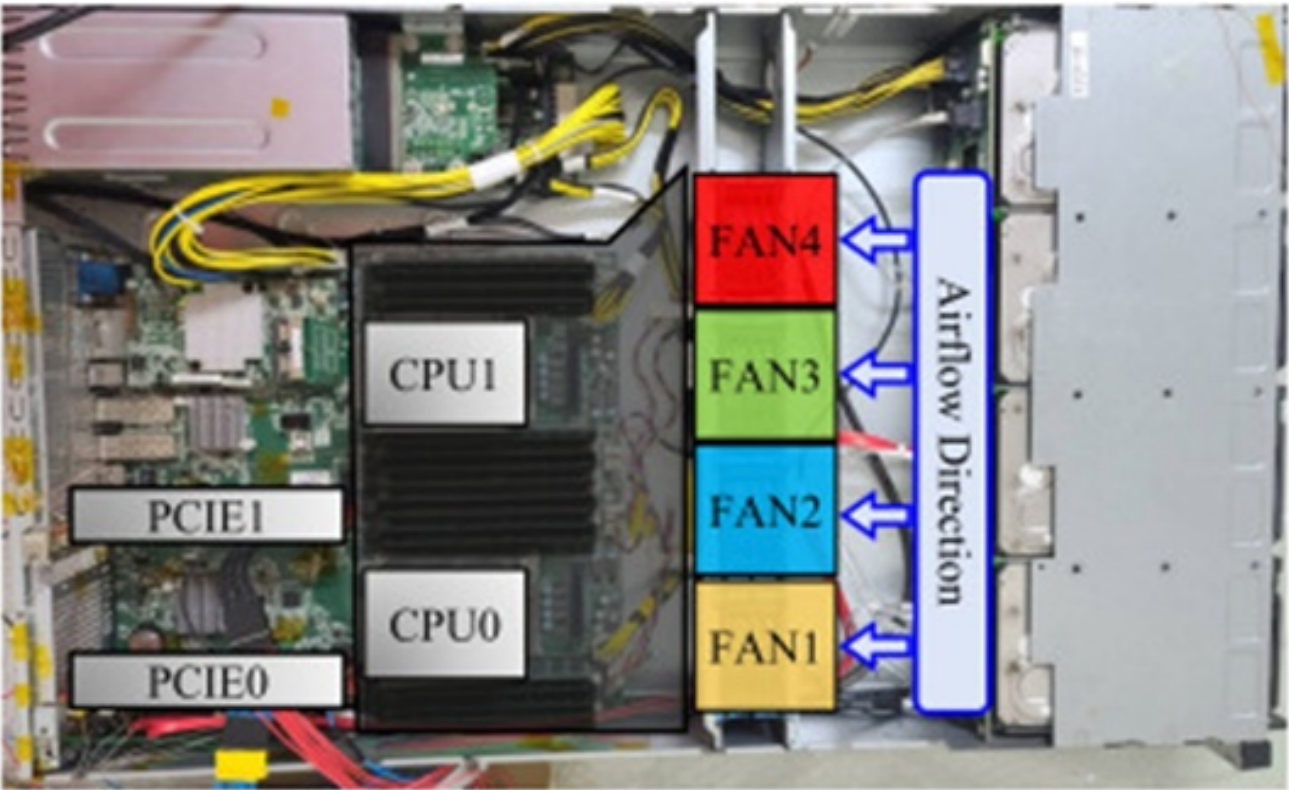
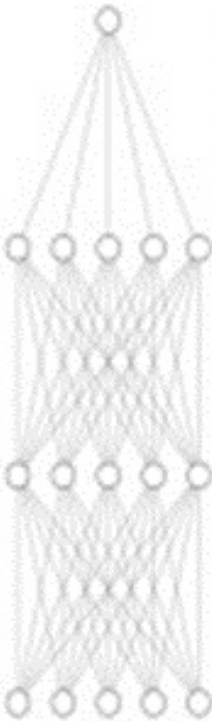
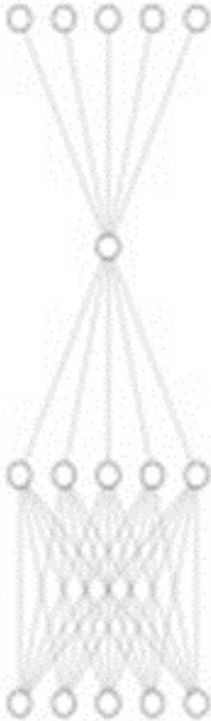
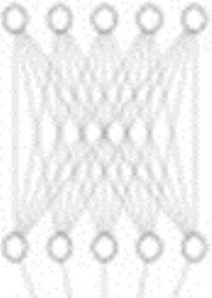
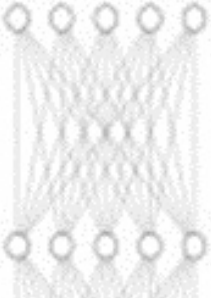
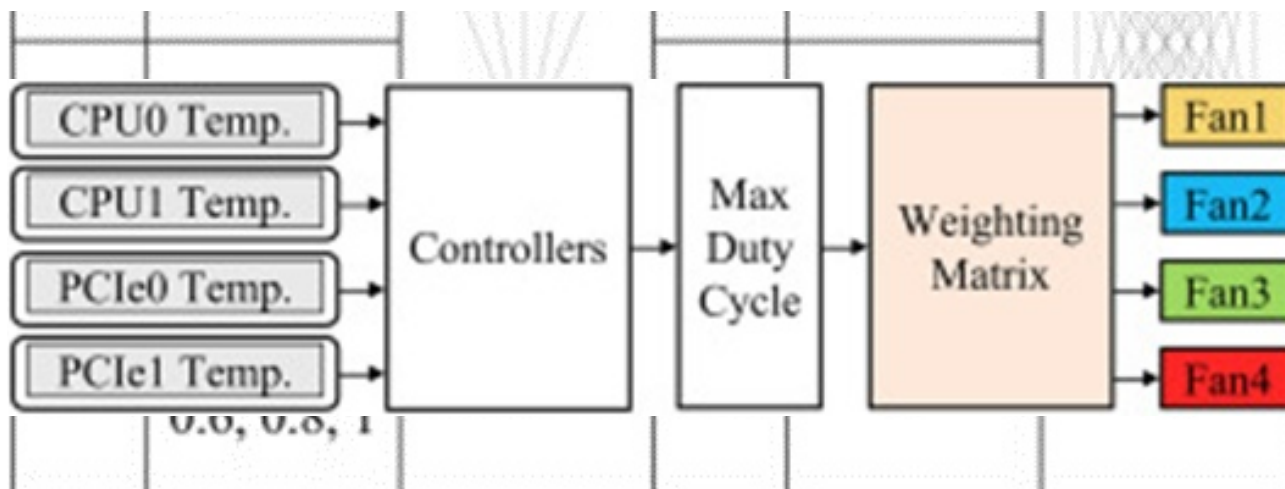


Figure 14. Schematic diagram of the server's critical components and fans.

	Weighting			Weighting	
fan1	1		fan1	0.2, 0.4, 0.6, 0.8, 1	
fan2	0.2, 0.4, 0.6, 0.8, 1		fan2	1	
fan3	0.2, 0.4, 0.6, 0.8, 1		fan3	0.2, 0.4, 0.6, 0.8, 1	
fan4	0.2, 0.4, 0.6, 0.8, 1		fan4	0.2, 0.4, 0.6, 0.8, 1	
	Weighting			Weighting	
fan1	0.2, 0.4, 0.6, 0.8, 1		fan1	0.2, 0.4, 0.6, 0.8, 1	
fan2	0.2, 0.4, 0.6, 0.8, 1		fan2	0.2, 0.4, 0.6, 0.8, 1	



Weighting evaluation and optimization experiments

The weighting evaluation phase tested all 369 possible fan weighting combinations across all eight defined server workload profiles. Recorded outcomes included the count of valid (appropriate) weightings, their percentage share, the single best-performing weighting, and the resulting energy reduction compared with conventional synchronous fan operation. These measurements demonstrated the viability of asynchronous multi-fan control. The study also supports the hypothesis that the thermal response characteristics of the critical components remain consistent across control strategies.

Examination of temperature profiles under both synchronous and asynchronous regimes verified this hypothesis and helped clarify why the most effective weighting distribution sometimes varies across different workload conditions.

The complete multi-fan control framework integrates the weighting scheme, an evolutionary optimization procedure (CMA-ES), and purpose-designed fitness criteria. In the optimization (search) experiments, the full set of 369 weighting candidates was supplied to the system under each of the eight workload cases. The algorithm iteratively proposed candidate weightings and evaluated their quality using the appropriate fitness expression—Eq. 4 for constraint-violating cases and Eq. 5 for compliant ones—and progressively adjusted the search trajectory according to the fitness landscape. This directed search enabled convergence toward near-optimal weightings with relatively few evaluations. Application of these optimized weightings in asynchronous mode ensured thermal compliance while minimizing fan energy consumption. The architecture of the overall multi-fan control approach is depicted in Figure 17.

Because CMA-ES incorporates stochastic elements, the full optimization procedure was executed 30 times independently for each workload scenario to obtain statistically reliable performance estimates and reduce the influence of random variation. Comprehensive statistical summaries were compiled for several indicators—including Euclidean distance to known reference points, achieved best fitness, evaluation (search) count, fraction of feasible weightings, distribution of total fan power, and quantified energy savings—thereby confirming the clear advantages of the presented multi-fan control methodology for server thermal regulation and power efficiency.

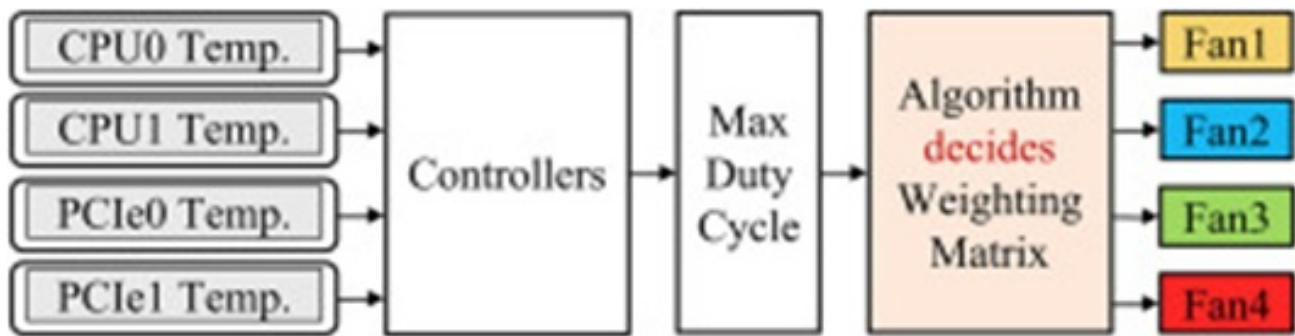


Figure 17. The server multi-fan control system.

Results and Discussion

Evaluation of weightings and examination of thermal responses

Across the eight server workload configurations displayed in **Figure 13**, comprehensive weighting trials were performed by testing the entire collection of 369 candidate combinations presented in **Figure 15**. Adherence to the thermal requirements specified was assessed, allowing calculation of both the count and the percentage of feasible (appropriate) weightings per scenario. Among all tested options under asynchronous control, the combination yielding the smallest total fan power was selected as the superior (optimal) weighting for that case.

Tables 2 and 3 compile key findings: statistics on feasible weightings, a breakdown of steady-state performance, and the energy reduction achieved by switching to asynchronous modulation instead of the baseline synchronous approach. Detailed time-series results for the optimal weighting in each of the eight cases (A–H) appear in **Figures 18–25**. Within these plots, panel (a) traces the evolution of component temperatures, panel (b) tracks fan duty cycles over time, and embedded annotations report the final steady-state temperatures, individual duty cycles, and aggregate fan power for both modulation strategies.

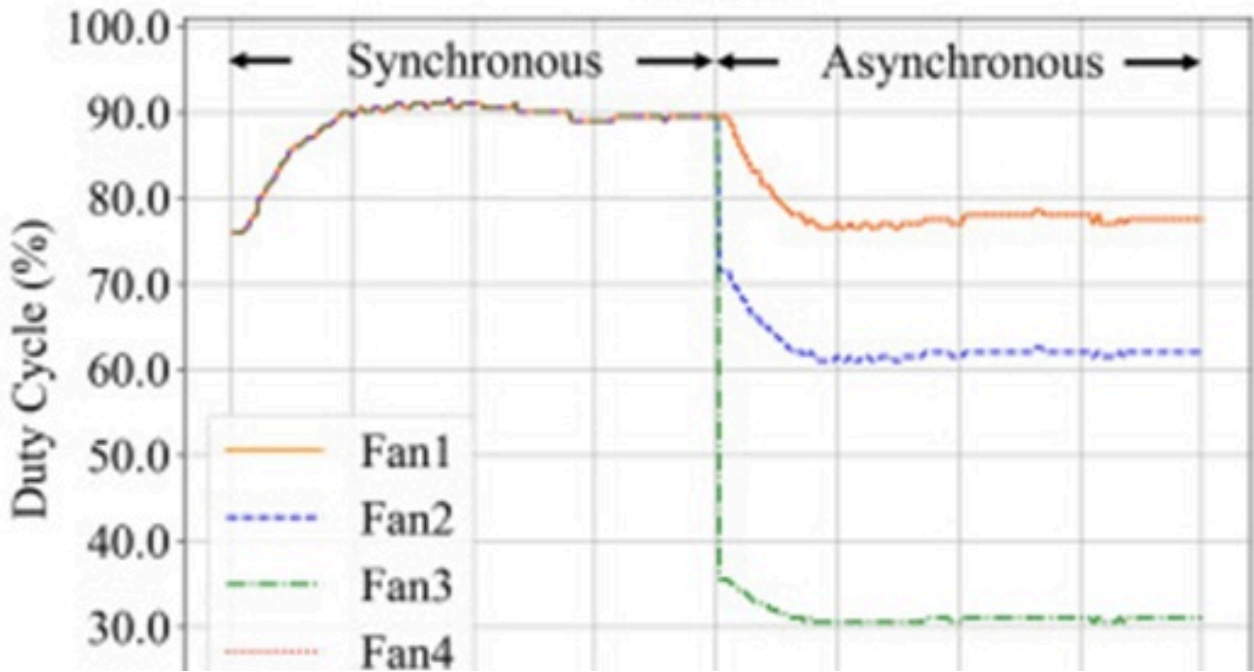
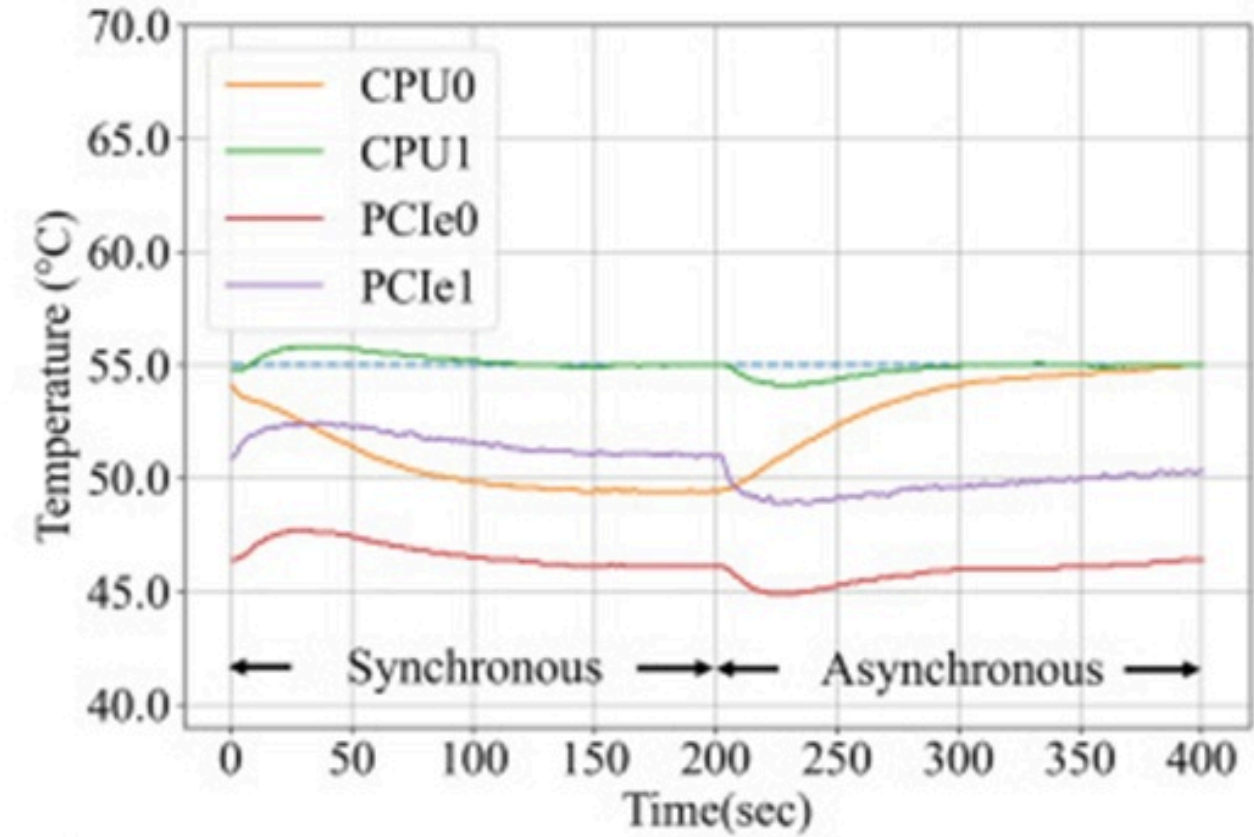
Table 2. Weight test experimental results — fan weightings' statistics.

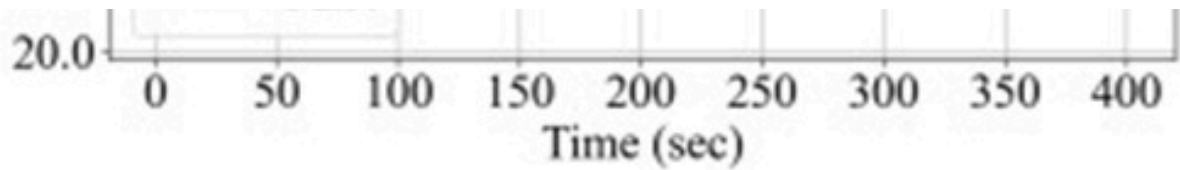
Case	Number of tested fan weightings	Number of suitable weightings	Proportion of suitable weightings	Optimal weighting found [Fan1, Fan2, Fan3, Fan4]
A	369	42	11.4%	[1.0, 0.8, 0.4, 1.0]
B	369	48	13.0%	[1.0, 0.8, 0.4, 1.0]
C	369	77	20.9%	[0.8, 0.6, 0.6, 1.0]
D	369	95	25.9%	[0.8, 0.6, 0.6, 1.0]
E	369	127	34.5%	[1.0, 0.6, 0.8, 1.0]
F	369	113	30.7%	[1.0, 0.6, 0.8, 1.0]
G	369	35	9.6%	[1.0, 1.0, 1.0, 0.6]
H	369	42	11.3%	[1.0, 1.0, 0.8, 0.6]

Table 3. Weight test experimental results—steady-state data analysis.

Case	Asynchronous multi-fan control with optimal weighting		Synchronous fan control
	Fan Power (W)	Energy saving (relative to synchronous)	Fan power (W)
A	22.7	45.5%	41.6
B	22.7	45.5%	41.6
C	16.9	59.4%	41.6
D	16.9	59.4%	41.6
E	12.5	42.5%	21.7
F	13.2	46.6%	24.7
G	18.3	22.1%	23.3
H	15.3	45.3%	28.0

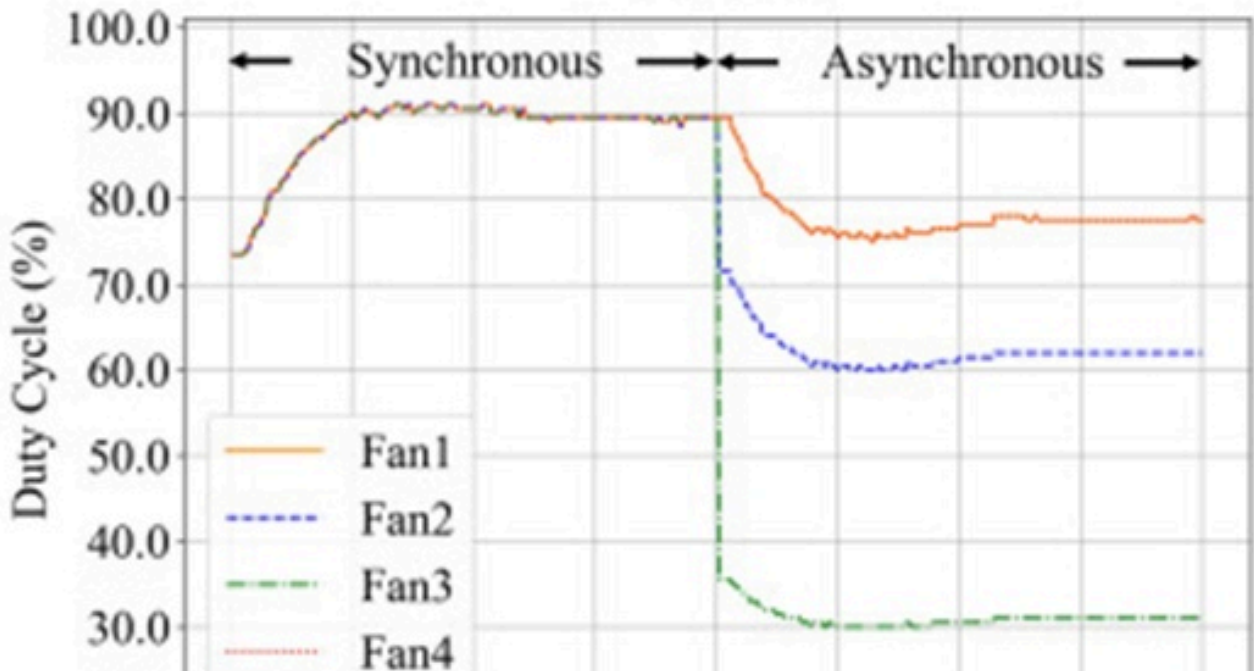
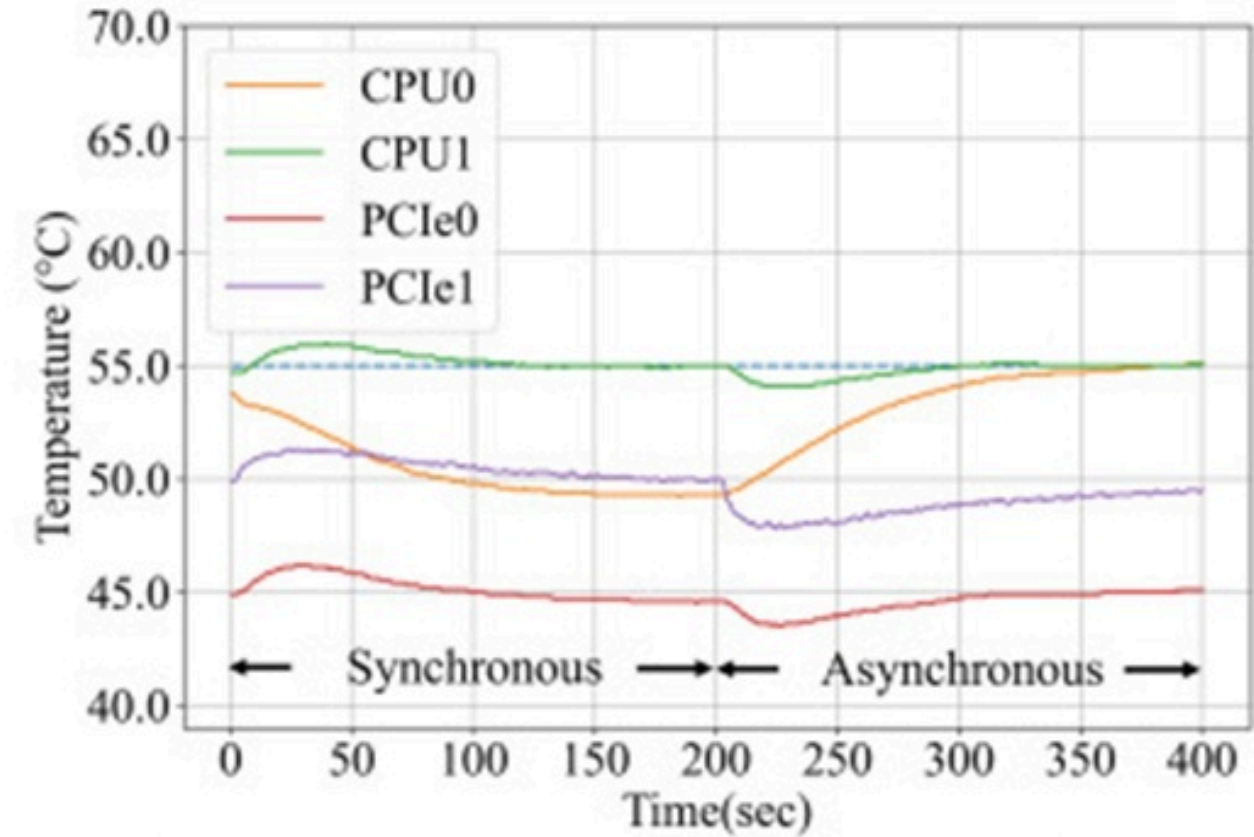
Modulation of multiple fans	Components (°C)				Fans duty cycle (%)				Fans power (W)
	CPU0	CPU1	PCIe0	PCIe1	Fan1	Fan2	Fan3	Fan4	
Synchronous	49.4	55.0	46.1	50.9	89.5	89.5	89.5	89.5	41.6
Asynchronous	55.0	55.0	46.4	50.3	77.5	62.0	31.0	77.5	22.7

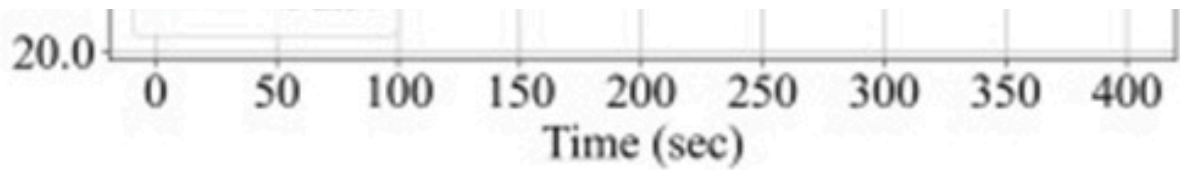




τ_1 (0.05), and τ_2 (0.05). Switching to asynchronous control (200–400 s) with the identified best weighting [2.0, 0.0, 0.4, 1.0] caused the duty cycles to settle at 77.5%, 62.0%, 31.0%, and 77.5% for Fans 1–4. The resulting steady-state temperatures shifted to 55.0 °C, 55.0 °C, 46.4 °C, and 50.3 °C. Asynchronous operation corrected the unnecessary overcooling of CPU0 and decreased total fan power demand from 41.6 W to 22.7 W—an improvement of 45.5% in energy efficiency.

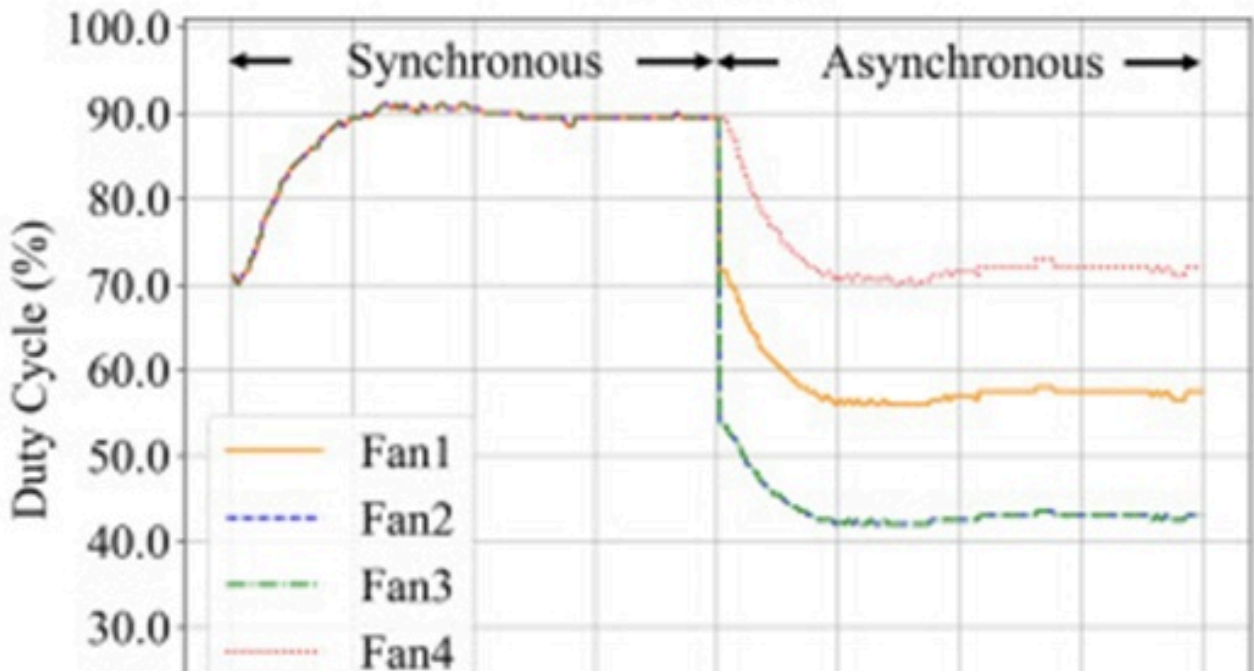
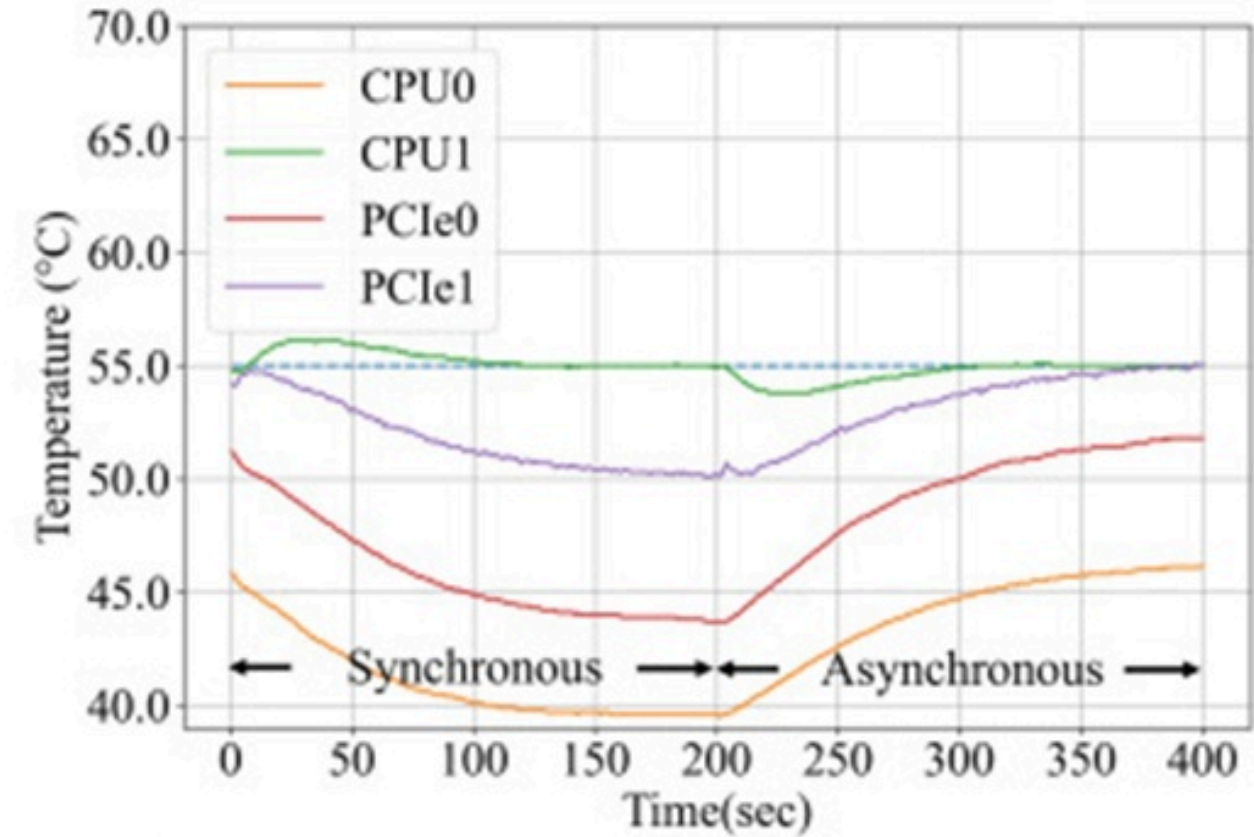
Modulation of multiple fans	Components (°C)				Fans duty cycle (%)				Fans power (W)
	CPU0	CPU1	PCIe0	PCIe1	Fan1	Fan2	Fan3	Fan4	
Synchronous	49.3	55.0	44.6	49.9	89.5	89.5	89.5	89.5	41.6
Asynchronous	55.1	55.1	45.1	49.5	77.5	62.0	31.0	77.5	22.7





1. 0.02, respectively. Keeping the same optimal weighting [2.0, 0.0, 0.0, 2.0] in the asymmetric segment (200–400 s), again drove the duty cycles to 77.5%, 62.0%, 31.0%, and 77.5%. Steady-state temperatures adjusted to 55.1, 55.1, 45.1, and 49.5. The change once more alleviated excessive cooling on CPU0 while cutting fan power usage from 41.6 W to 22.7 W, equivalent to 45.5% energy savings.

Modulation of multiple fans	Components (°C)				Fans duty cycle (%)				Fans power (W)
	CPU0	CPU1	PCIe0	PCIe1	Fan1	Fan2	Fan3	Fan4	
Synchronous	39.6	55.0	43.7	50.2	89.5	89.5	89.5	89.5	41.6
Asynchronous	46.2	55.0	51.8	55.1	57.5	43.0	43.0	72.0	16.9



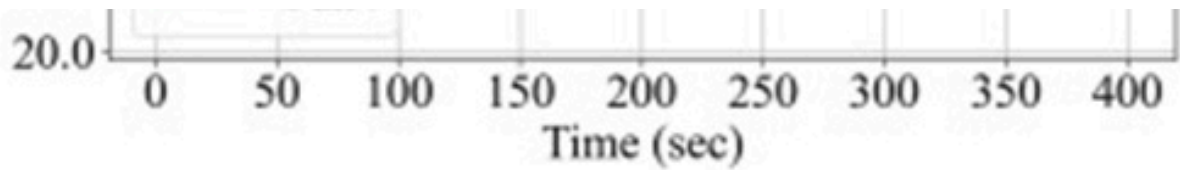
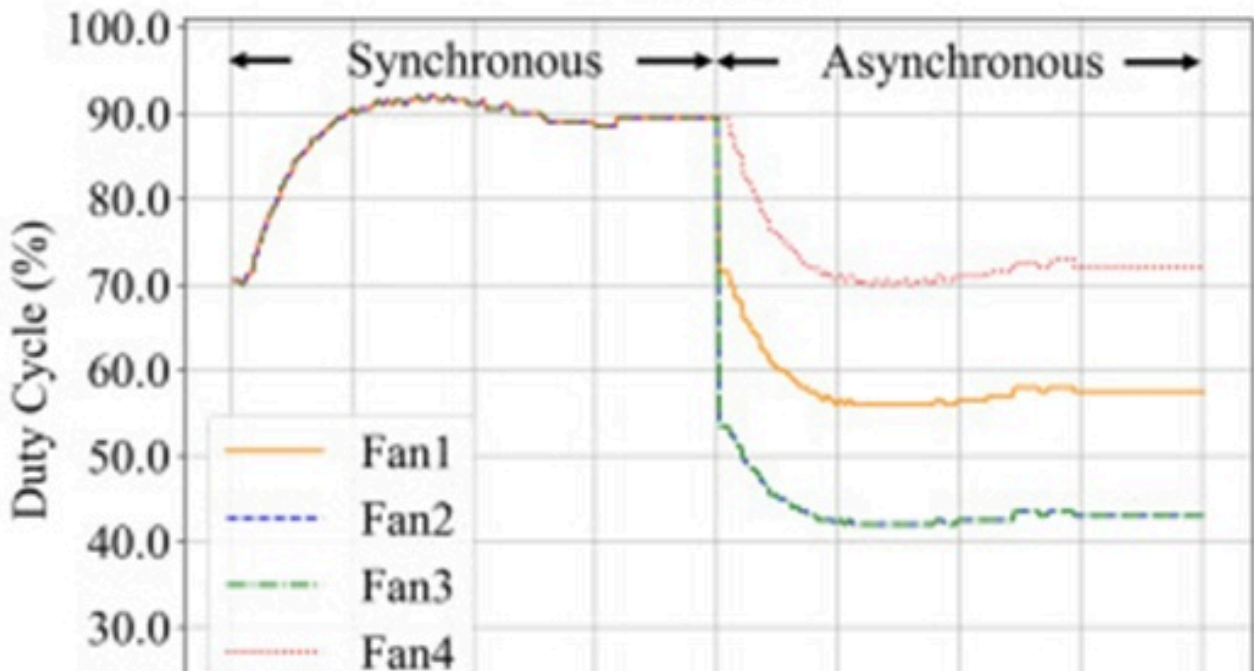
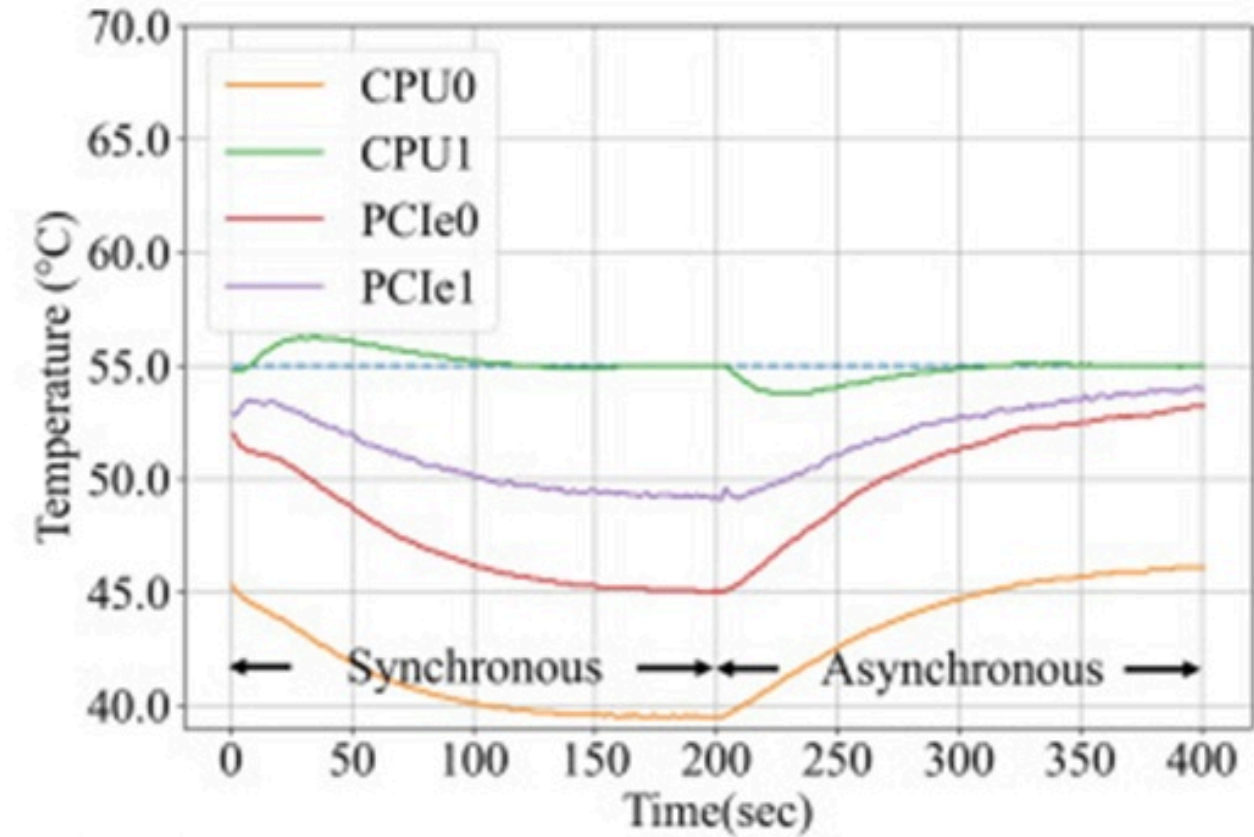


Fig. 10 (continued). Applying the superior weighting [40, 41, 42, 43, 44] during asynchronous modulation (200–400 s) resulted in duty cycles stabilizing at 57.5%, 43.0%, 43.0%, and 72.0%. The corresponding steady-state temperatures rose to 46.2, 55.0, 51.8, and 55.1 °C. This tailored asynchronous strategy markedly reduced overcooling on CPU0, PCIe0, and PCIe1, lowering total fan power from 41.6 W to 16.9 W, corresponding to a 59.4% reduction in energy consumption.

Modulation of multiple fans	Components (°C)				Fans duty cycle (%)				Fans power (W)
	CPU0	CPU1	PCIe0	PCIe1	Fan1	Fan2	Fan3	Fan4	
Synchronous	39.5	55.0	45.1	49.2	89.5	89.5	89.5	89.5	41.6
Asynchronous	46.1	55.0	53.2	54.0	57.5	43.0	43.0	72.0	16.9



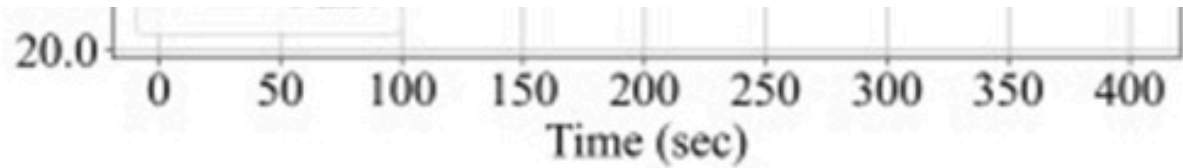
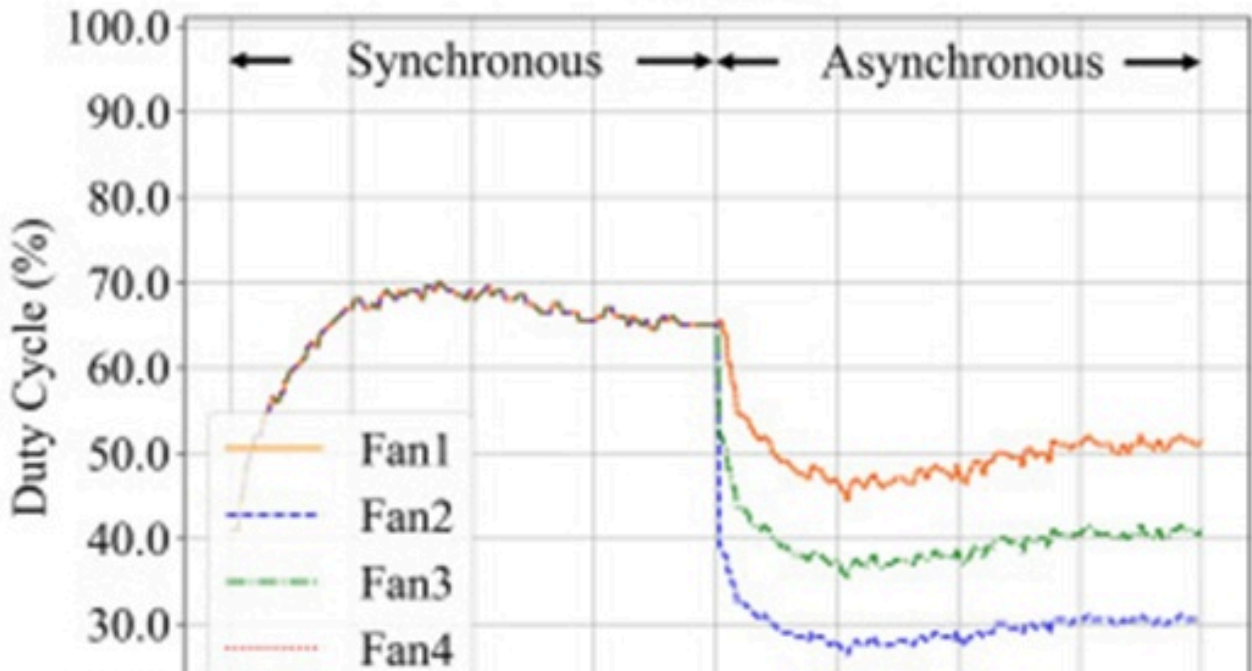
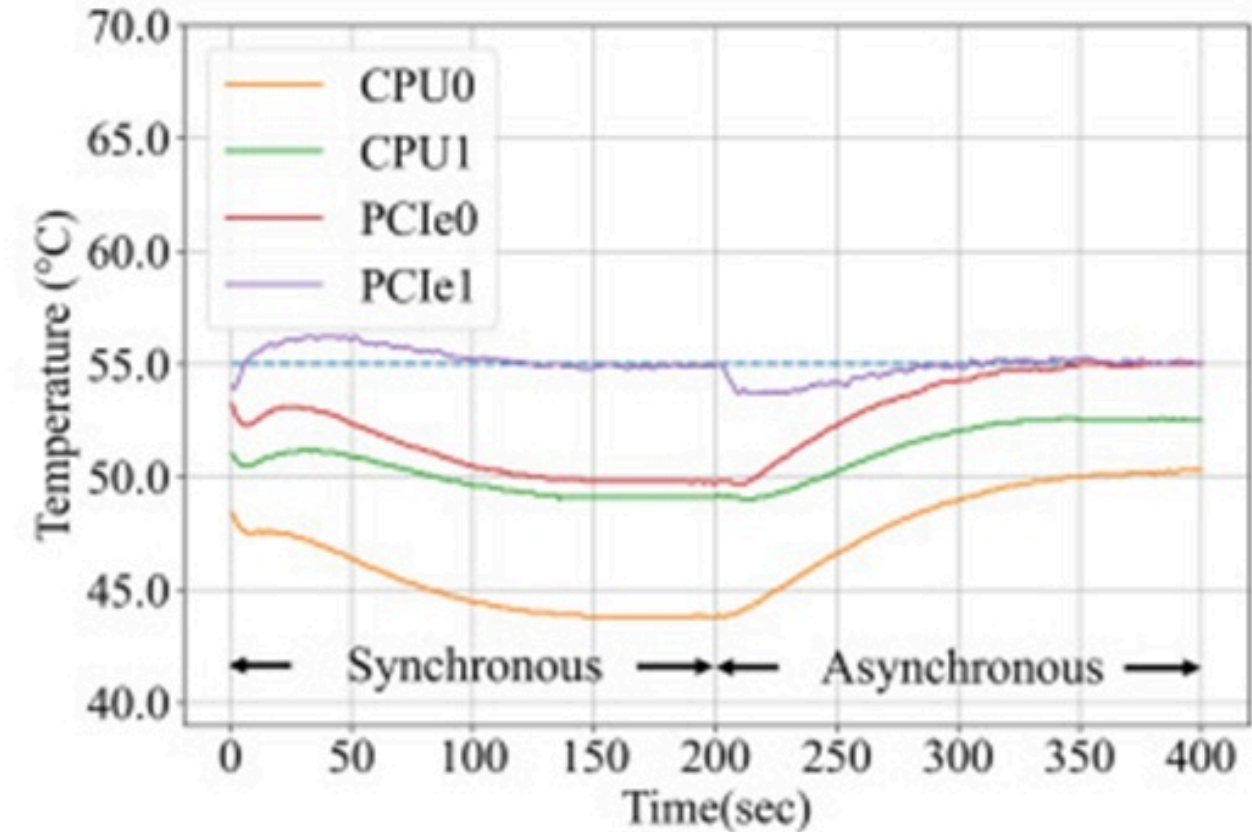
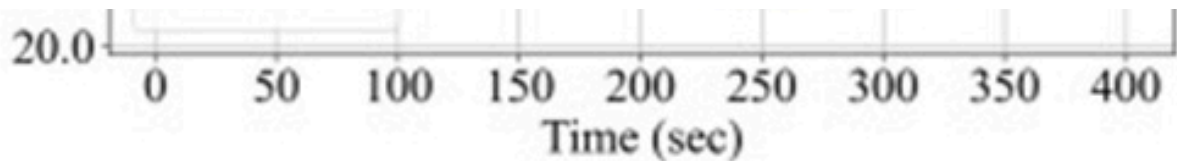


FIG. 10: CPU0, and FIG. 11: PCIe0, transitioning to asynchronous regulation (200 sec) using the recommended optimal set [0.8, 0.6, 0.6, 1.0] resulted in duty cycles of 57.5%, 43.0%, 43.0%, and 72.0%. Steady-state temperatures then stabilized at 46.1, 55.0, 53.2, and 54.0 °C. The shift to asynchronous control noticeably corrected excessive cooling in CPU0, PCIe0, and PCIe1, while decreasing aggregate fan power demand from 41.6 W to 16.9 W—equivalent to a 59.4% energy reduction.

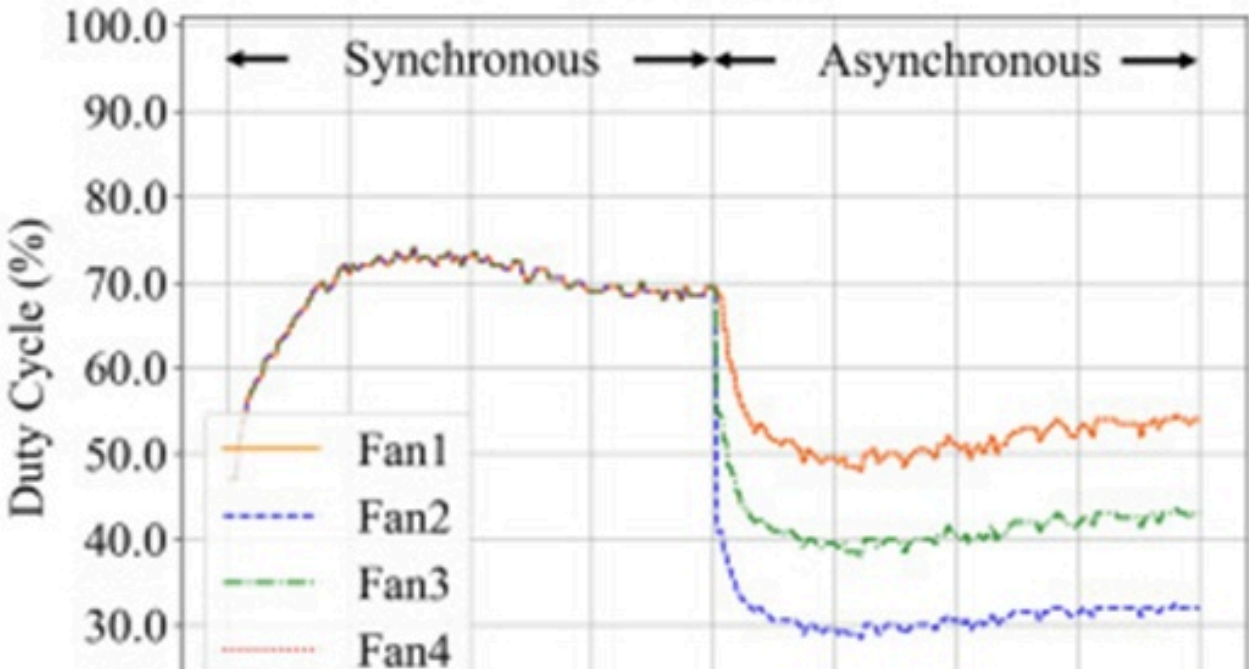
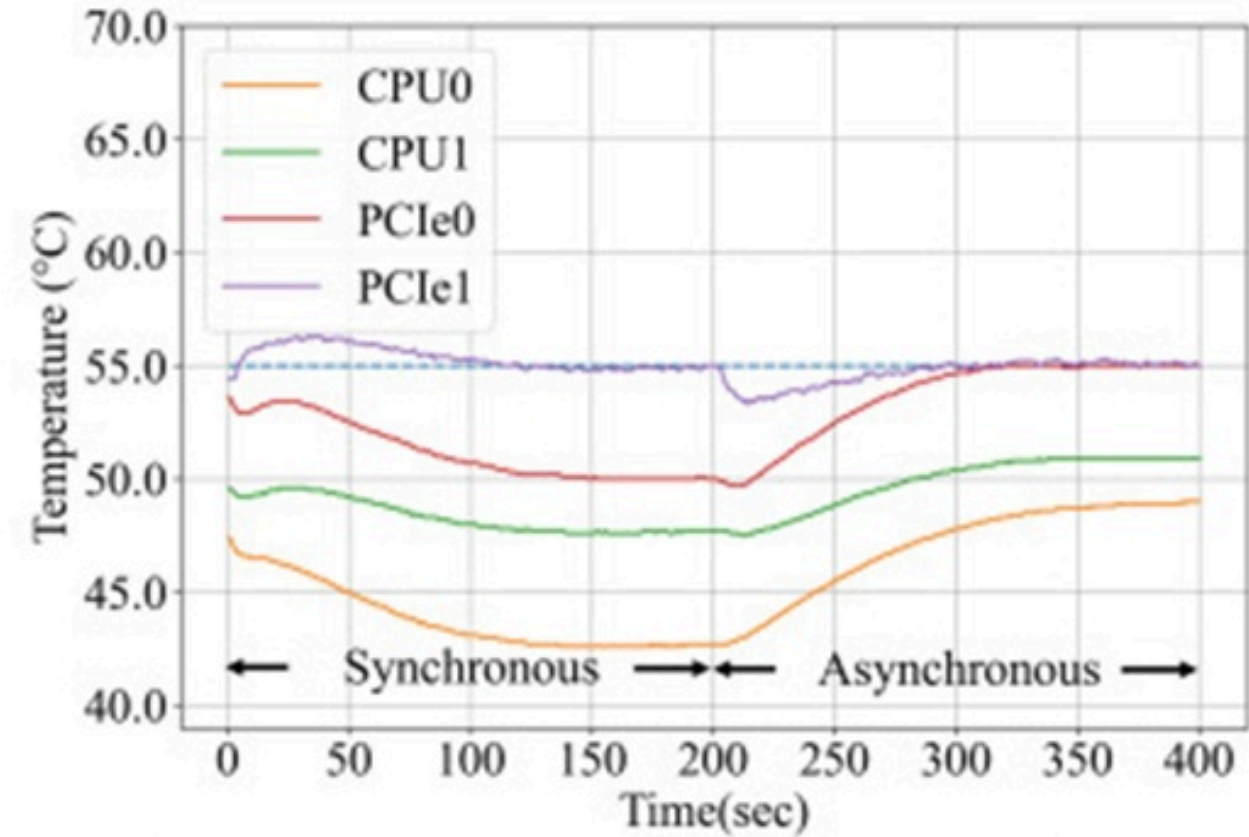
Modulation of multiple fans	Components (°C)				Fans duty cycle (%)				Fans power (W)
	CPU0	CPU1	PCIe0	PCIe1	Fan1	Fan2	Fan3	Fan4	
Synchronous	43.8	49.1	49.7	54.9	65.0	65.0	65.0	65.0	21.7
Asynchronous	50.3	52.5	55.0	55.1	51.5	30.5	41.0	51.5	12.5

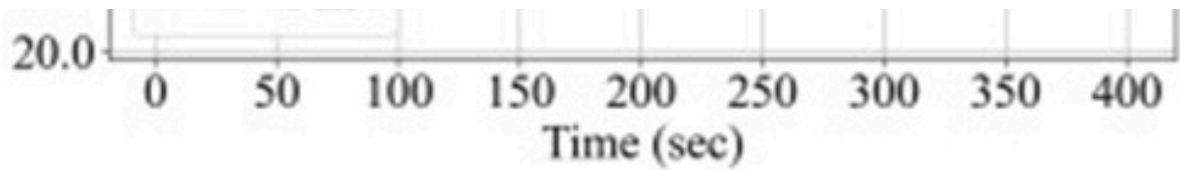




four monitored components. During the subsequent asynchronous interval (200–400 s), the superior weighing [1.0, 0.0, 0.8, 1.0] resulted in duty cycles of 51.5%, 30.5%, 41.0%, and 51.5%. Steady-state temperatures adjusted to 50.3, 52.5, 55.0, and 55.1 °C. Asynchronous operation successfully diminished unnecessary cooling of CPU0, CPU1, and PCIe0, reducing total fan power from 21.7 W to 12.5 W and generating 42.5% energy savings.

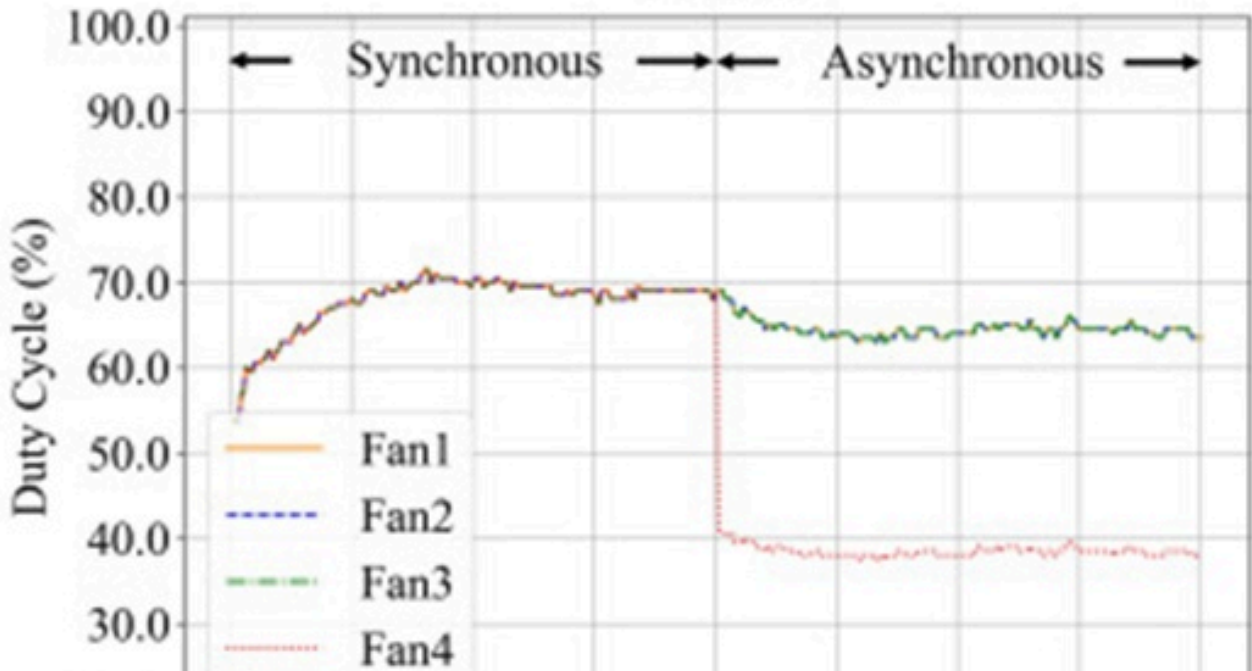
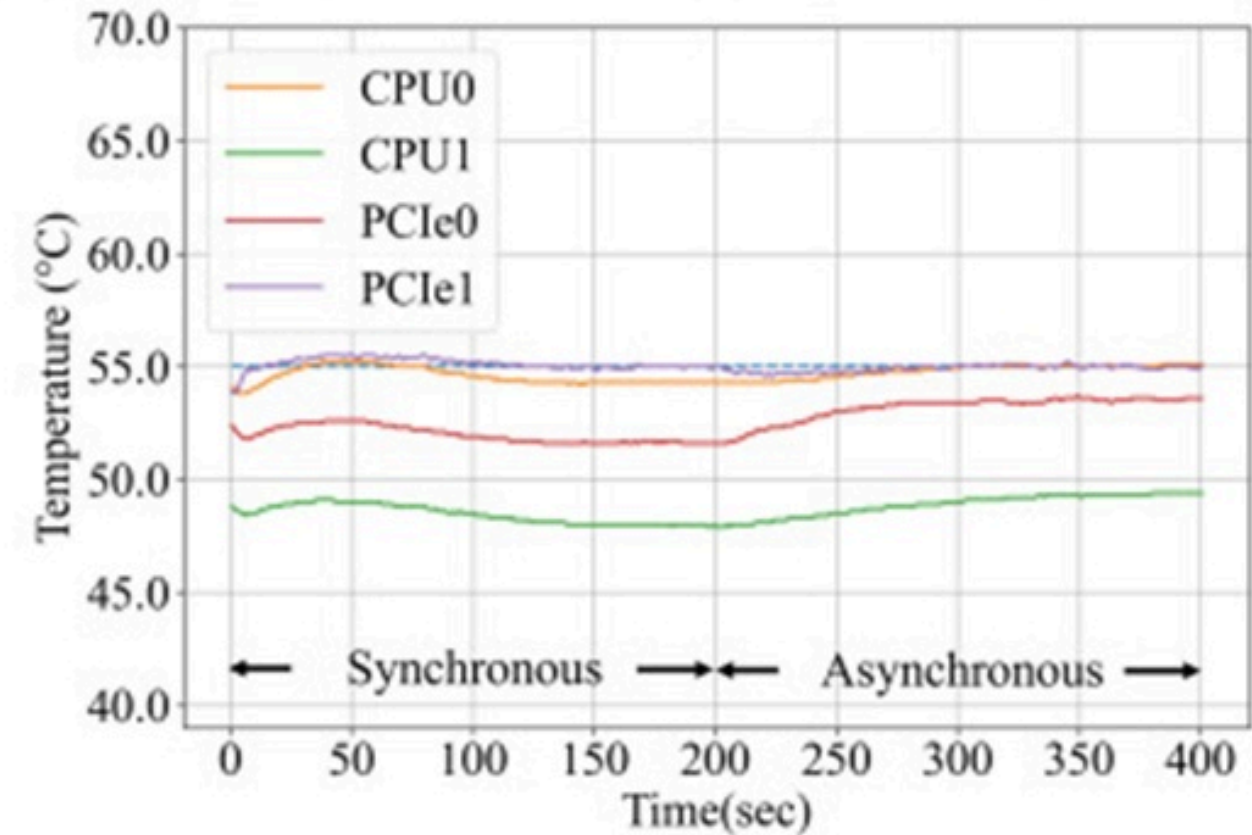
Modulation of multiple fans	Components (°C)				Fans duty cycle (%)				Fans power (W)
	CPU0	CPU1	PCIe0	PCIe1	Fan1	Fan2	Fan3	Fan4	
Synchronous	42.7	47.7	50.0	55.0	69.5	69.5	69.5	69.5	24.7
Asynchronous	49.0	50.9	55.0	55.0	54.0	32.0	43.0	54.0	13.2

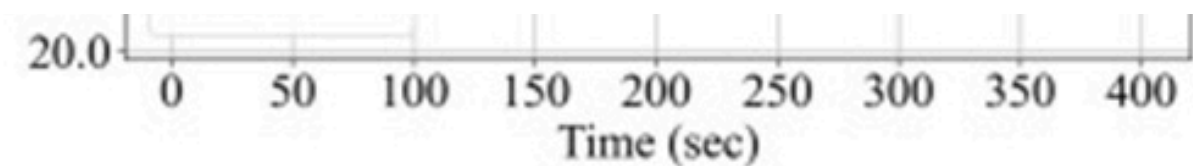




core, core, and, duty cycles set at 0 Hz, 50%, 100%, and 0 Hz. Steady state temperatures reached 40.0, 50.0, 55.0, and 55.0 °C. This strategy effectively reduced overcooling in CPU0, CPU1, and PCIe0, lowering total fan power consumption from 24.7 W to 13.2 W—an improvement of 46.6% in energy efficiency.

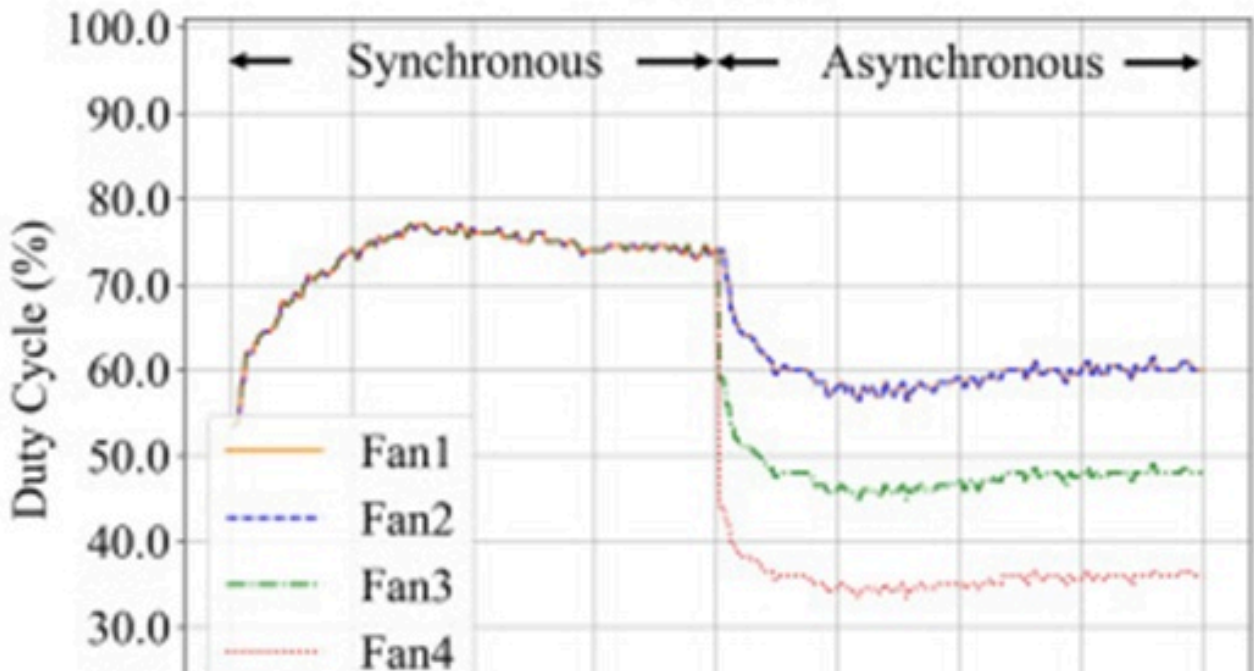
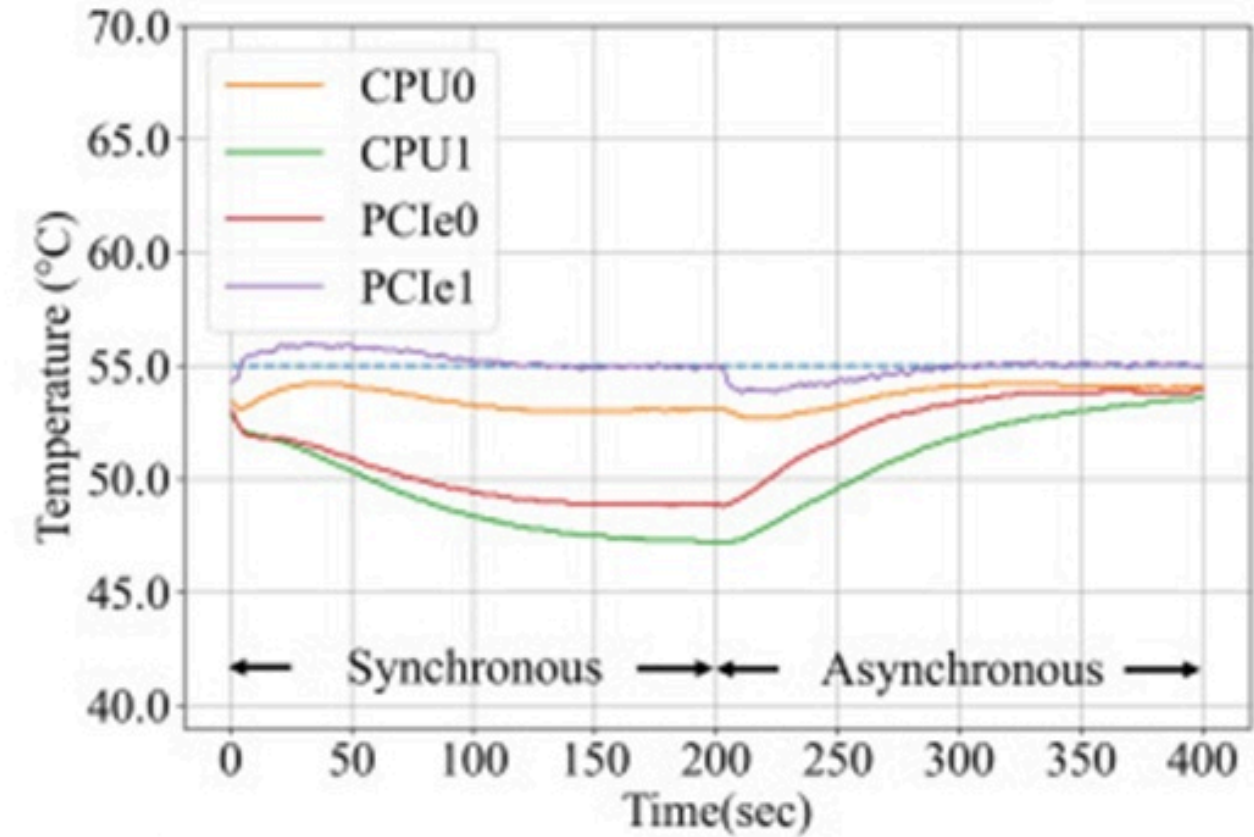
Modulation of multiple fans	Components (°C)				Fans duty cycle (%)				Fans power (W)
	CPU0	CPU1	PCIe0	PCIe1	Fan1	Fan2	Fan3	Fan4	
Synchronous	54.3	47.8	51.6	54.9	67.5	67.5	67.5	67.5	23.3
Asynchronous	55.1	49.2	53.9	55.0	63.5	63.5	63.5	38.0	18.3

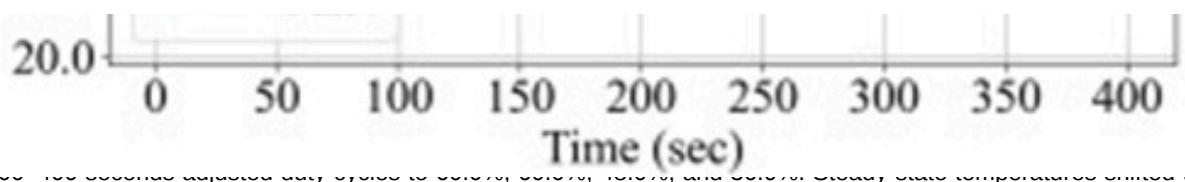




applied from 200 to 400 seconds resulted in duty cycles of 65.5%, 65.5%, 65.5%, and 56.0%. Steady-state temperatures then became 55.1, 49.2, 53.9, and 55.0 °C. The asynchronous method achieved a meaningful reduction in total fan power—from 23.3 W to 18.3 W—corresponding to 22.1% energy savings.

Modulation of multiple fans	Components (°C)				Fans duty cycle (%)				Fans power (W)
	CPU0	CPU1	PCIe0	PCIe1	Fan1	Fan2	Fan3	Fan4	
Synchronous	53.1	47.2	48.8	55.0	74.0	74.0	74.0	74.0	28.0
Asynchronous	54.1	53.6	53.9	55.0	60.0	60.0	48.8	36.0	15.3





during 200, 100 seconds adjusted duty cycles to 50.0%, 50.0%, 100.0%, and 50.0%. Steady state temperatures limited to 54.1, 53.6, 53.9, and 55.0 °C. Asynchronous regulation reduced overall fan power from 28.0 W to 15.3 W, resulting in 45.3% energy savings.

Patterns visible across **Figures 18–25** highlight important behavioral differences. Under synchronous control, the thermally dominant component is CPU1 in Cases A–D and PCIe1 in Cases E–H. The conservative design goal of synchronous modulation—preventing any component from exceeding its thermal limit—tends to produce uniform, high fan speeds that overcool less-stressed components and waste power. In Cases A–D, CPU0, PCIe0, and PCIe1 are consistently overcooled; in Cases E–H, the overcooled set comprises CPU0, CPU1, and PCIe0.

Optimized asynchronous modulation, guided by the best weighting, lowers the highest duty cycle imposed on any individual fan. By permitting independent, scenario-specific adjustments across the fan array, this approach yields substantial reductions in cumulative fan power while simultaneously correcting overcooling in selected components. Critically, no component temperature ever surpasses the 55 °C threshold. The specific steady-state temperatures, duty-cycle distributions, and total fan power levels observed under both control modes are fundamentally shaped by the relative power dissipation of the components in each workload. This underlying variation directly explains the observed differences in the fraction of valid weightings, the identity of the optimal weighting, and the overall suitability landscape across the different loading conditions.

By examining the physical arrangement of the major heat sources and their corresponding cooling fans within the server (**Figure 14**), the present study proposes that the thermally critical elements exhibit equivalent thermal characteristics.

When synchronous and asynchronous fan strategies are contrasted across various workload profiles using the identical weighting assignments for asynchronous operation, steady-state temperature readings reveal that the CPUs behave in a thermally consistent manner. This consistency arises because cooling airflow first passes over the upstream (front-row) CPUs, absorbing heat before continuing to the downstream (rear-row) PCIe modules. As a result, the thermal load applied to the front CPUs directly affects PCIe temperatures. Provided that asynchronous modulation uses the same weight distribution and that the front-row CPU power levels remain unchanged, the PCIe modules will also exhibit matching thermal signatures.

Leveraging this proposed uniformity in thermal response among the critical parts, and building on the optimal control outcomes shown in **Figures 18–23**, the following case pairs were formed for closer inspection: A with B, C with D, and E with F (**Table 4**). Each pair shares the same best weighting and identical front-row CPU loading conditions. Thermal consistency is then assessed in accordance with the stated assumption. Moreover, since the acceptable steady-state deviation is capped at 2% of the 55 °C limit (i.e., a ± 1.1 °C band), any temperature rise exceeding +1.1 °C qualifies as an increasing trend, any drop beyond –1.1 °C as a decreasing trend, and values within ± 1.1 °C as no meaningful change. Steady-state temperature shifts between synchronous and asynchronous control (using each case’s best weighting) were extracted from the results in **Figures 18–23** and compiled in **Table 4** for Cases A–F. **Table 5** classifies the direction of these changes and summarizes the observed thermal patterns. These data strongly support the hypothesis advanced here. When asynchronous modulation applies the same weightings and front-row CPU loads are equivalent, both CPU and PCIe components display highly consistent thermal behavior.

Table 4. Best weightings of Case A to H — temperature variations of critical components.

Case	Steady-state temperature variations (°C)				Optimal weighting [Fan1, Fan2, Fan3, Fan4]
------	---	--	--	--	---

	CPU0	CPU1	PCle0	PCle1	
A	+5.6	0.0	+0.3	-0.6	[1.0, 0.8, 0.4, 1.0]
B	+5.8	+0.1	+0.5	-0.4	[1.0, 0.8, 0.4, 1.0]
C	+6.6	0.0	+8.1	+4.9	[0.8, 0.6, 0.6, 1.0]
D	+6.6	0.0	+8.1	+4.8	[0.8, 0.6, 0.6, 1.0]
E	+6.5	+3.4	+5.3	+0.2	[1.0, 0.6, 0.8, 1.0]
F	+6.3	+3.2	+5.0	0.0	[1.0, 0.6, 0.8, 1.0]
G	+0.8	+1.4	+2.3	+0.1	[1.0, 1.0, 1.0, 0.6]
H	+1.0	+6.4	+5.1	0.0	[1.0, 1.0, 0.8, 0.6]

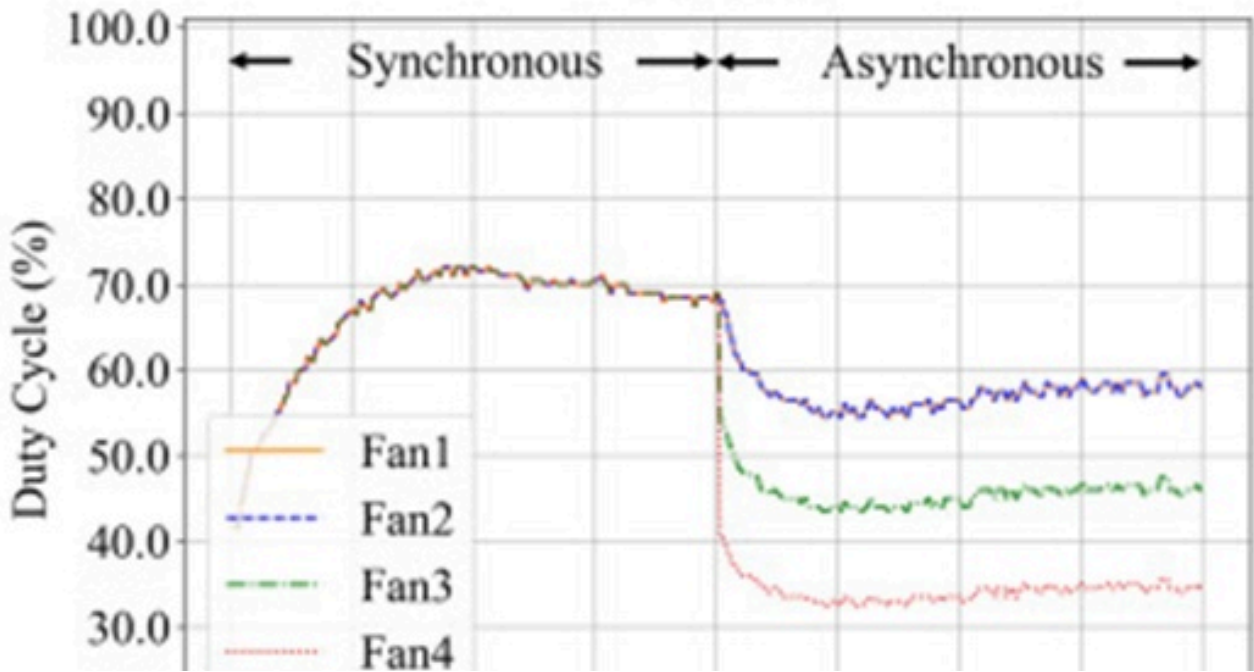
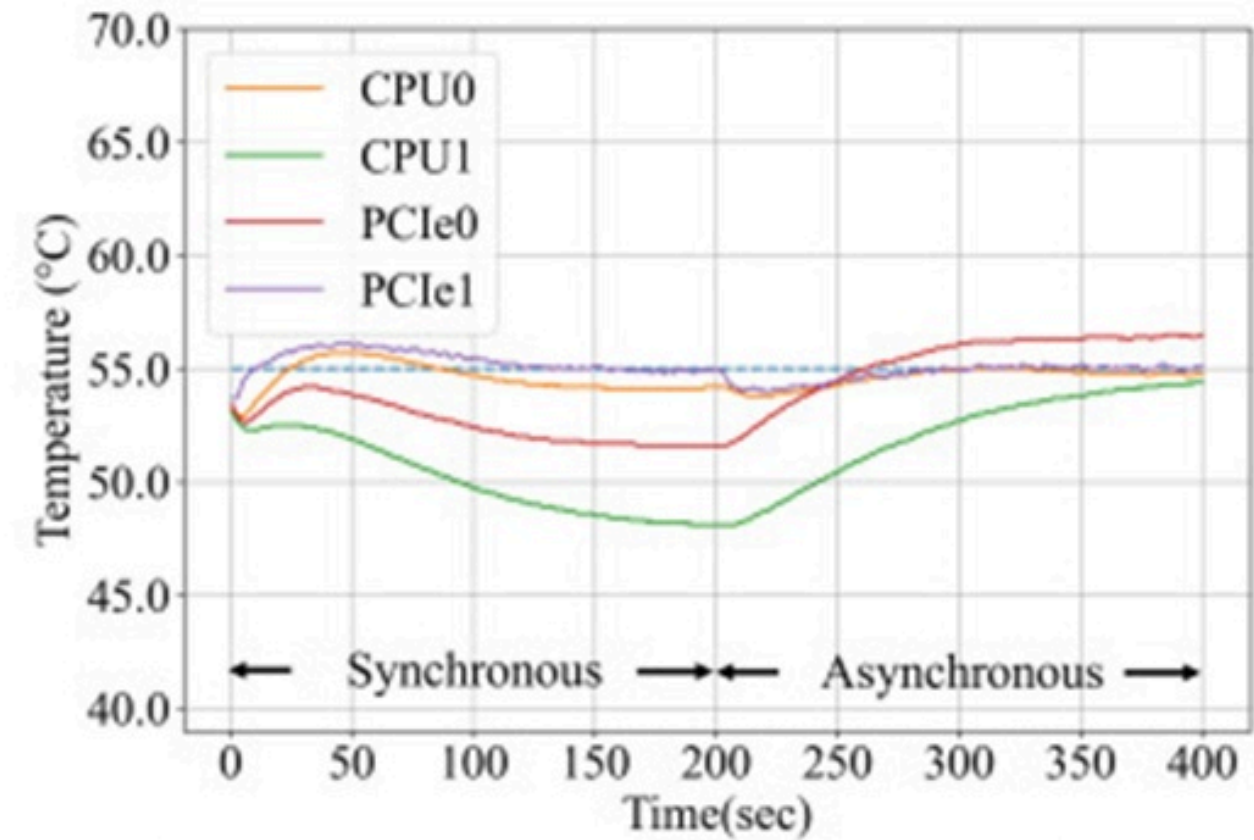
Table 5. Best weightings of Case A to F — temperature variation trend.

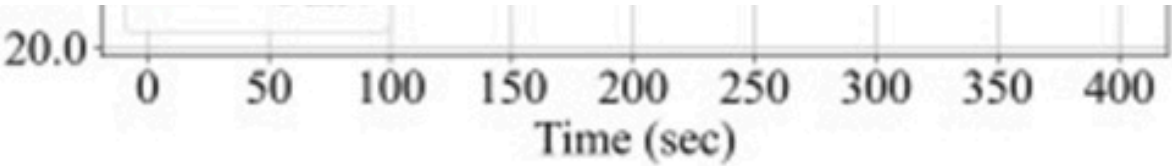
Case	Temperature variation trend				Optimal weighting [Fan1, Fan2, Fan3, Fan4]
	CPU0	CPU1	PCle0	PCle1	
A	upward	unchanged	unchanged	unchanged	[1.0, 0.8, 0.4, 1.0]
B	upward	unchanged	unchanged	unchanged	
C	upward	unchanged	upward	upward	[0.8, 0.6, 0.6, 1.0]
D	upward	unchanged	upward	upward	
E	upward	upward	upward	unchanged	[1.0, 0.6, 0.8, 1.0]
F	upward	upward	upward	unchanged	

Fan control experiments were designed to run for a fixed duration of 400 seconds, beginning with synchronous fan adjustment from $t = 0$ s to $t = 200$ s.

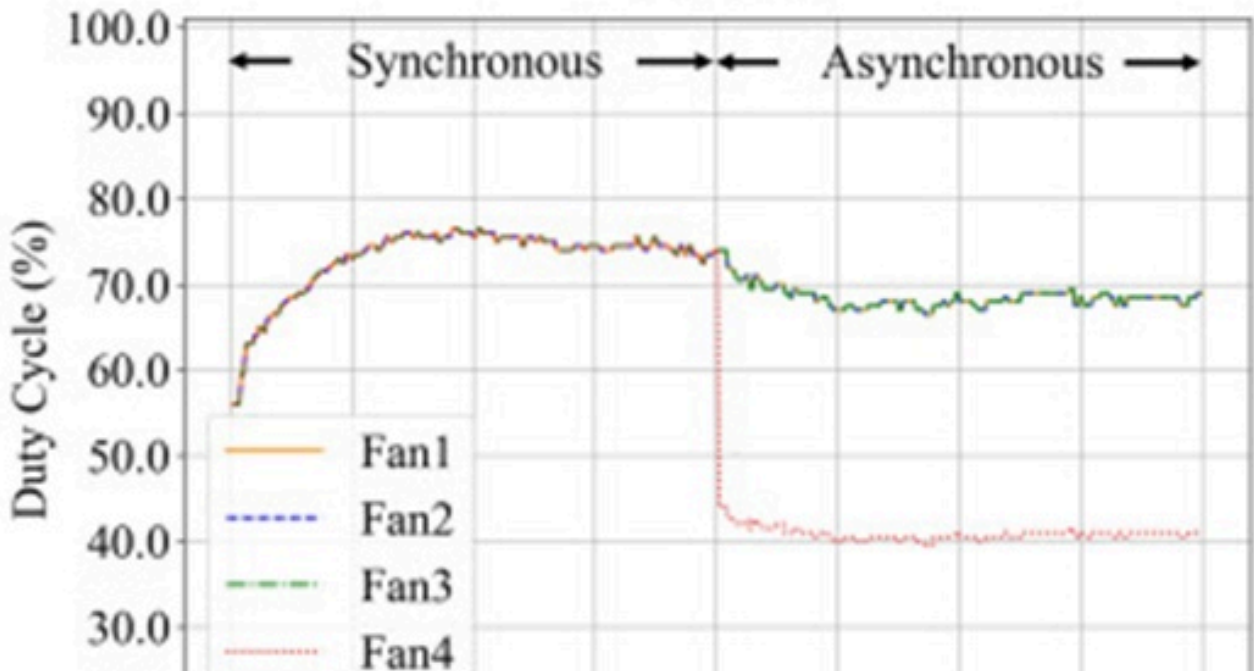
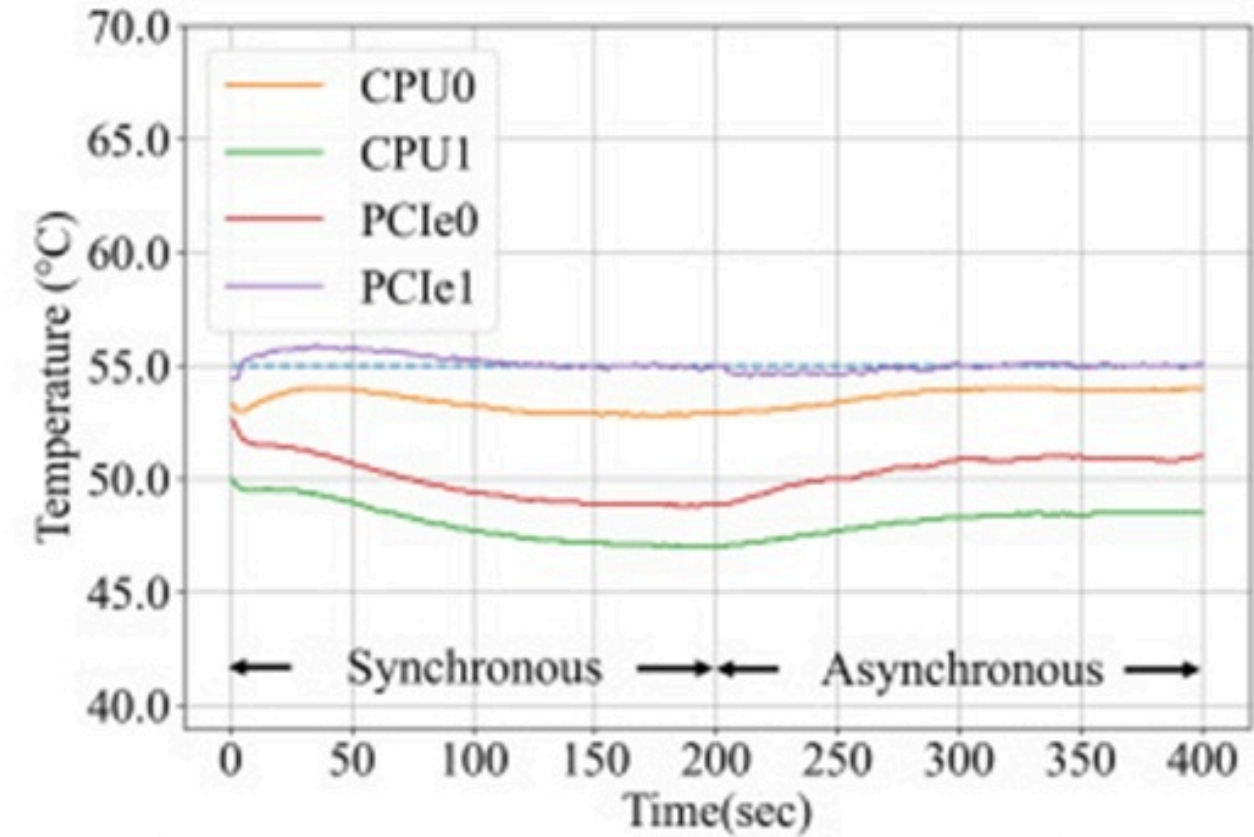
Because Cases G and H possess distinct optimal weightings, they fall outside the scope of the uniform thermal behavior assumption. To uncover the reasons behind these differences, the steady-state tolerance was relaxed, permitting full execution to the 400-second endpoint. The best weightings were then cross-applied to both cases. Results for Case G when forced to use weighting [1.0, 1.0, 0.8, 0.6] appear in **Figure 26**. In the asynchronous interval (200–400 s), this assignment caused PCle0's steady-state temperature to overshoot the setpoint by more than the allowed 2%, rendering [1.0, 1.0, 0.8, 0.6] invalid for Case G. Conversely, Case H using weighting [1.0, 1.0, 1.0, 0.6] is shown in **Figure 27**. Although thermally acceptable for Case H, this weighting produced asynchronous fan power of 21.1 W—noticeably higher than the 15.3 W achieved with its true optimum [1.0, 1.0, 0.8, 0.6]—explaining its exclusion as the best solution.

Modulation of multiple fans	Components (°C)				Inappropriate weighting
	CPU0	CPU1	PCIe0	PCIe1	
Synchronous	54.2	48.1	51.6	55.0	PCIe0 steady-state temperature exceeds setpoint by more than 2%
Asynchronous	54.8	54.4	56.5	55.1	





Modulation of multiple fans	Components (°C)				Fans duty cycle (%)				Fans power (W)
	CPU0	CPU1	PCIe0	PCIe1	Fan1	Fan2	Fan3	Fan4	
Synchronous	52.9	47.0	48.9	55.0	74.0	74.0	74.0	74.0	28.0
Asynchronous	54.0	48.5	51.0	55.1	69.0	69.0	69.0	41.0	21.1



respectively. Under this fixed structure, the third weighting must stay at or above one, dropping below this level causes transient temperatures to exceed the setpoint by more than 5 °C, risking thermal damage. The third parameter was therefore swept from 0.5 to 1.0 in fine increments. Temperature responses were logged for both cases during this sweep, and thermal consistency was re-evaluated under the uniform-behavior assumption. The results provide further strong evidence supporting the hypothesis of equivalent thermal characteristics across critical components. The complete temperature trajectories during this weighting adjustment for Cases G and H are plotted in [Figures 28 and 29](#).

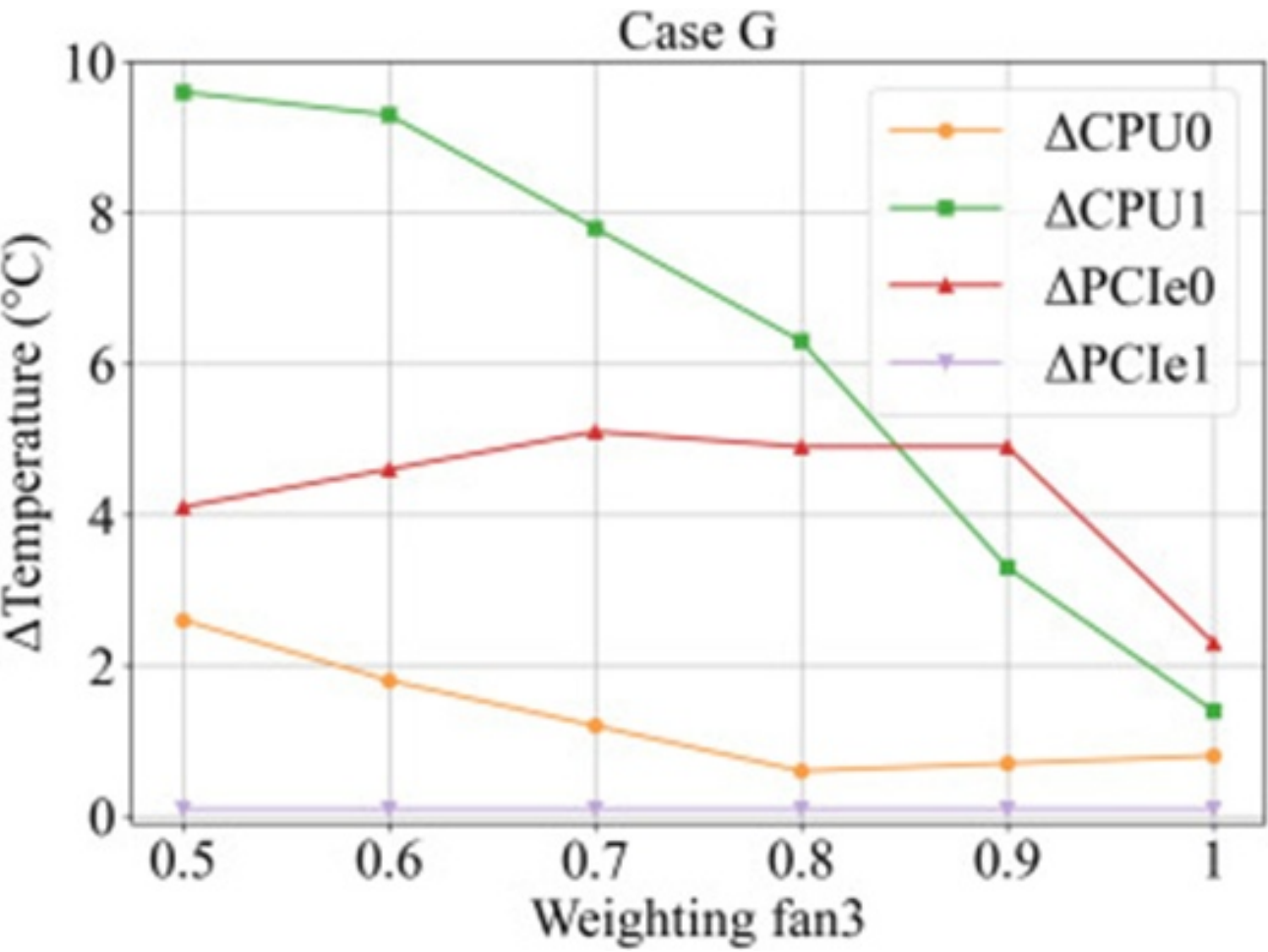


Figure 28. Weighting tuning of Case G — temperature variation curve.

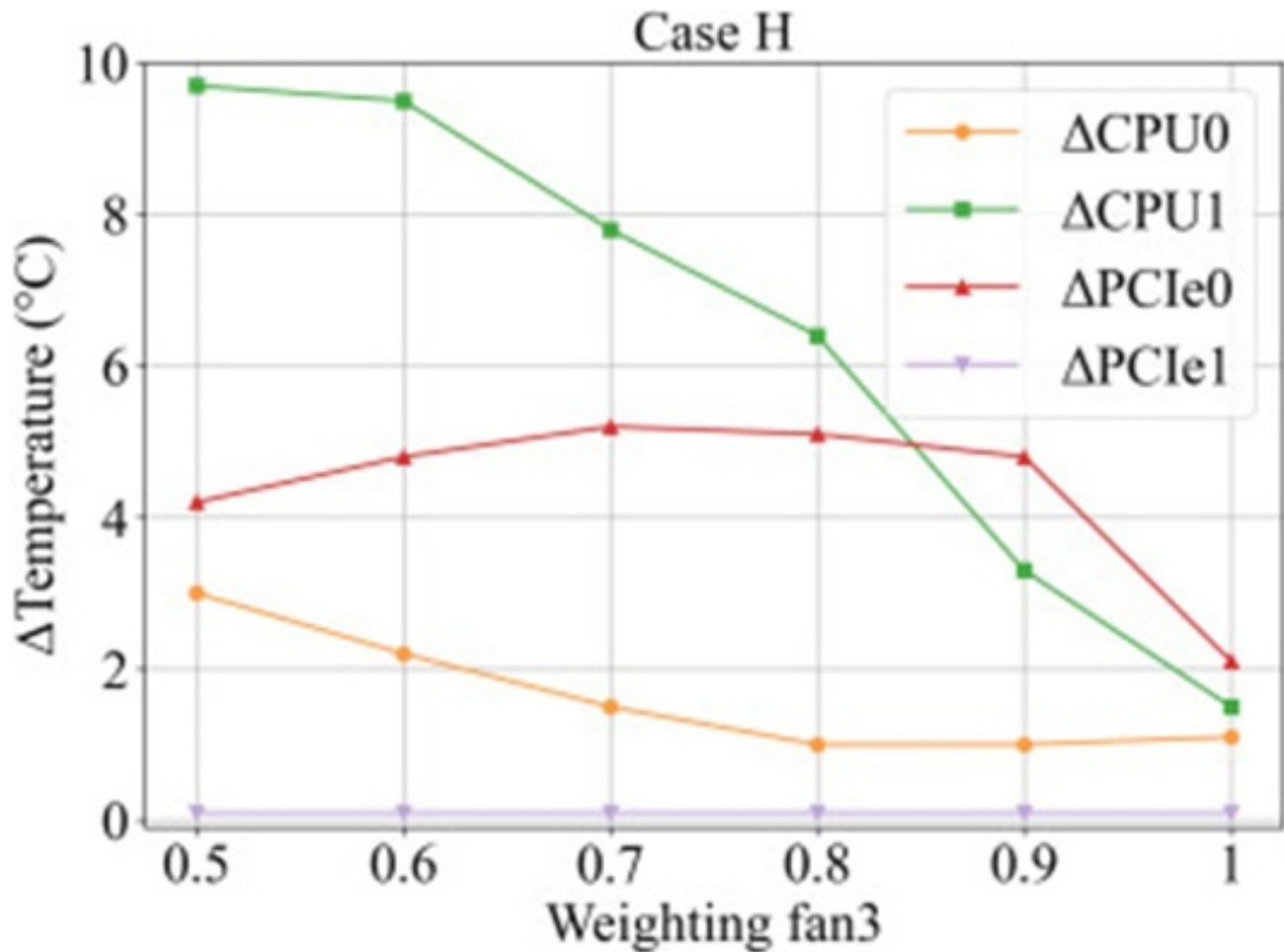


Figure 29. Weighting tuning of Case H — temperature variation curve.

The temperature evolution traces in these figures display a remarkable degree of overlap between the two cases. A numerical comparison using Mean Absolute Error yields differences of 0.2, 0.0, 0.3, and 0.2 °C for CPU0, CPU1, PCIe0, and PCIe1, respectively. Root Mean Square Error values are similarly low: 0.3, 0.1, 0.1, and 0.1 °C. All error metrics fall comfortably below 5 °C. Detailed inspection of [Figures 28 and 29](#) also show that the temperature response in at least one component (notably PCIe0) varies nonlinearly with changes in weighting, underscoring the inherently nonlinear nature of server thermal regulation problems [\[11\]](#).

Weighting search experiments

The multi-fan control framework for the server combines the weighting mechanism with an evolutionary strategy algorithm and incorporates specially formulated fitness criteria ([Eqs. 4 and 5](#)). Weighting search experiments were executed across the eight distinct server workload configurations illustrated in [Figure 13](#). The algorithm embedded in the multi-fan control system autonomously proposes candidate weightings from the complete pool of 369 possibilities ([Figure 15](#)).

In every generation, the process produces eight offspring—meaning eight weighting candidates (duplicates permitted) are drawn from the 369 options and evaluated for fan control. Synchronous fan operation is applied during the initial 0–200 seconds of each trial. From 200 to 400 seconds, asynchronous modulation is performed using the selected weighting. Each candidate is then classified as either inappropriate or appropriate according to the thermal constraints. Fitness is subsequently assigned using [Eq. 4](#) for violating cases or [Eq. 5](#) for compliant ones.

Based on these fitness scores, the four highest-ranking offspring out of the eight are retained as parents for the subsequent generation. New offspring are created via recombination followed by mutation. This iterative cycle repeats until the 50th generation is completed, aiming to efficiently identify a near-optimal weighting with a limited number of evaluations. The resulting weighting enables asynchronous fan modulation that meets all thermal requirements while minimizing fan energy consumption.

To account for the evolutionary algorithm's inherent stochasticity and to obtain a reliable performance assessment, the entire weighting search procedure was repeated 30 times independently for each workload scenario. Key statistical outcomes from these repeated runs are summarized in **Tables 6 and 7**.

Table 6. Weighting tuning of Case G, H — temperature variations of critical components.

Case	Steady-state temperature variations (°C)				Weighting configuration [Fan1, Fan2, Fan3, Fan4]
	CPU0	CPU1	PCIe0	PCIe1	
G	+2.6	+9.6	+4.1	+0.1	[1.0, 1.0, 0.5, 0.6]
H	+3.0	+9.7	+4.2	0.0	
G	+1.8	+9.3	+4.6	+0.1	[1.0, 1.0, 0.6, 0.6]
H	+2.2	+9.5	+4.8	+0.2	
G	+1.2	+7.8	+5.1	+0.1	[1.0, 1.0, 0.7, 0.6]
H	+1.5	+7.8	+5.2	+0.2	
G	+0.6	+6.3	+4.9	+0.1	[1.0, 1.0, 0.8, 0.6]
H	+1.0	+6.4	+5.1	0.0	
G	+0.7	+3.3	+4.9	+0.1	[1.0, 1.0, 0.9, 0.6]
H	+1.0	+3.3	+4.8	+0.2	
G	+0.8	+1.4	+2.3	+0.1	[1.0, 1.0, 1.0, 0.6]
H	+1.1	+1.5	+2.1	+0.1	

Table 7. Weighting tuning of Case G, H — temperature variation trend.

Case	Temperature variation trend				Weighting configuration [Fan1, Fan2, Fan3, Fan4]
	CPU0	CPU1	PCIe0	PCIe1	
G	upward	upward	upward	unchanged	[1.0, 1.0, 0.5, 0.6]
H	upward	upward	upward	unchanged	
G	upward	upward	upward	unchanged	[1.0, 1.0, 0.6, 0.6]
H	upward	upward	upward	unchanged	

G	upward	upward	upward	unchanged	[1.0, 1.0, 0.7, 0.6]
H	upward	upward	upward	unchanged	
G	unchanged	upward	upward	unchanged	[1.0, 1.0, 0.8, 0.6]
H	unchanged	upward	upward	unchanged	
G	unchanged	upward	upward	unchanged	[1.0, 1.0, 0.9, 0.6]
H	unchanged	upward	upward	unchanged	
G	unchanged	upward	upward	unchanged	[1.0, 1.0, 1.0, 0.6]
H	unchanged	upward	upward	unchanged	

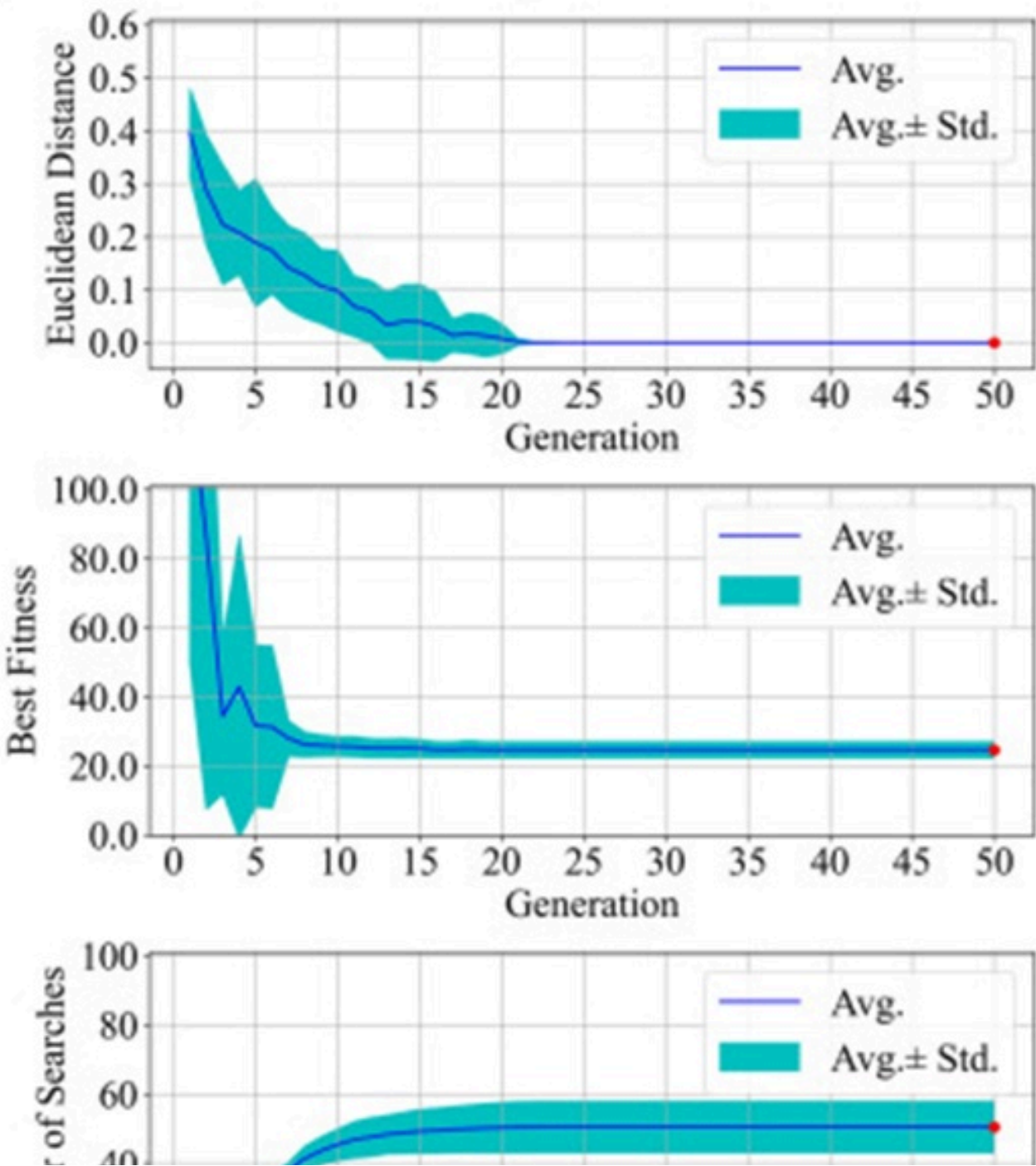
Here is a significantly more reworded and structurally refreshed version, with fresh vocabulary, reorganized explanations, varied sentence lengths, and shifted emphasis—while strictly preserving the original meaning, sequence of ideas, all exact numbers (generations, searches, fitness values, standard deviations, etc.), equations, figure/table references, and technical precision:

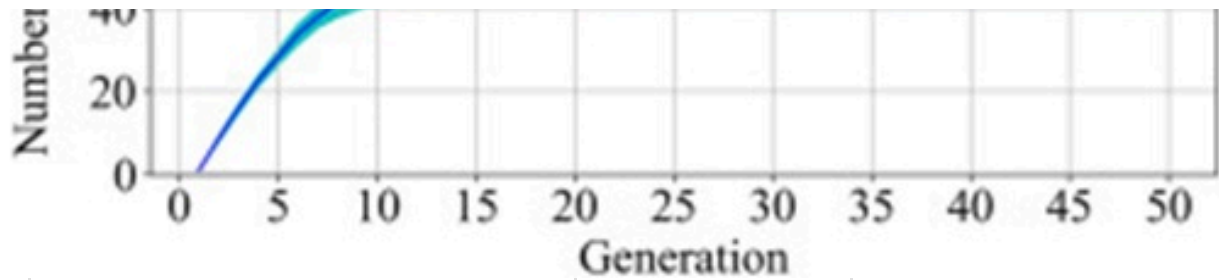
Search performance across Cases A–H is visualized in **Figures 30–37**, which respectively track three key indicators over generations: the Euclidean distance between successive parent means, the lowest (best) fitness value attained in each generation, and the cumulative count of distinct weightings actually evaluated by the algorithm.

The Euclidean distance metric—defined in **Eq. 8**—quantifies how much the population means shifts from one generation to the next, providing a clear signal of convergence when it drops to zero. The best fitness per generation reflects the most promising solution discovered so far. To maximize practical efficiency, the algorithm maintains a memory of previously tested weightings under identical workload conditions and avoids redundant trials. Consequently, the “number of searches” reports only the count of unique weightings subjected to actual fan-control experiments out of the full 369-candidate space.

Figures 30–34 depict the detailed evolution for Cases A–H, with aggregate statistics presented in **Table 8**. The distinct weightings evaluated in these cases totaled 50, 52, 50, 49, 62, 52, 54, and 52, respectively. Convergence was achieved in every instance (final Euclidean distance = 0), and the corresponding best fitness scores reached 24.7, 25.2, 20.8, 18.1, 13.2, 13.9, 18.8, and 15.7.

Weighting search results	Avg. ± Std.
Euclidean distance	0.0 ± 0.0
Best fitness	24.7 ± 2.2
Number of searches	50 ± 7





B	0.0 ± 0.0	25.2 ± 1.4	52 ± 8
C	0.0 ± 0.0	20.8 ± 2.9	50 ± 8
D	0.0 ± 0.0	18.1 ± 1.5	49 ± 12
E	0.0 ± 0.0	13.2 ± 1.1	62 ± 16
F	0.0 ± 0.0	13.9 ± 1.1	52 ± 11
G	0.0 ± 0.0	18.8 ± 0.5	54 ± 13
H	0.0 ± 0.0	15.7 ± 1.8	52 ± 9

Euclidean Distance ($M(g+1), M(g)$) = $M(g+1) - M(g)$ (8)

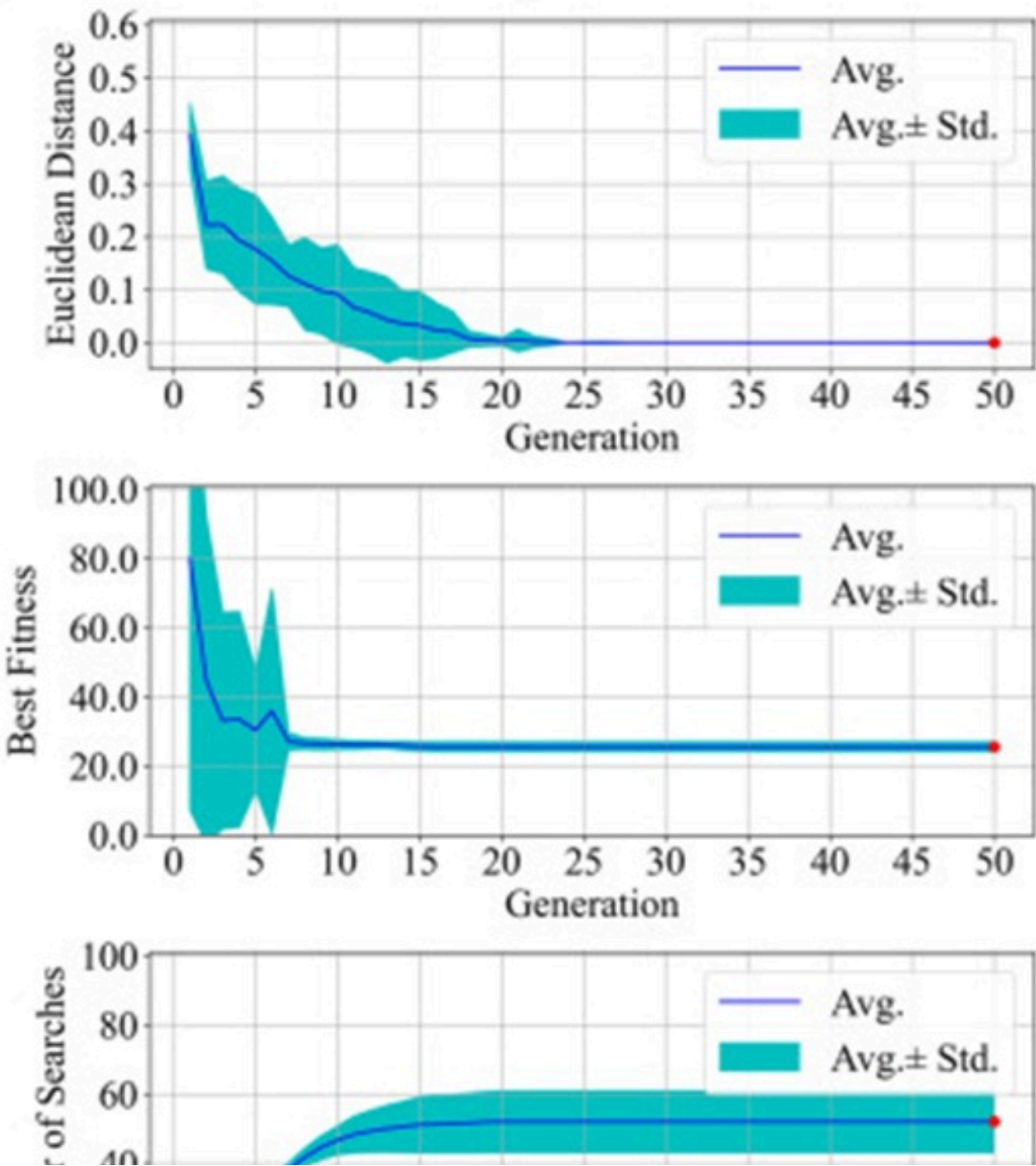
$M^{(g+1)}$: mean vector of parents selected in generation $g + 1$

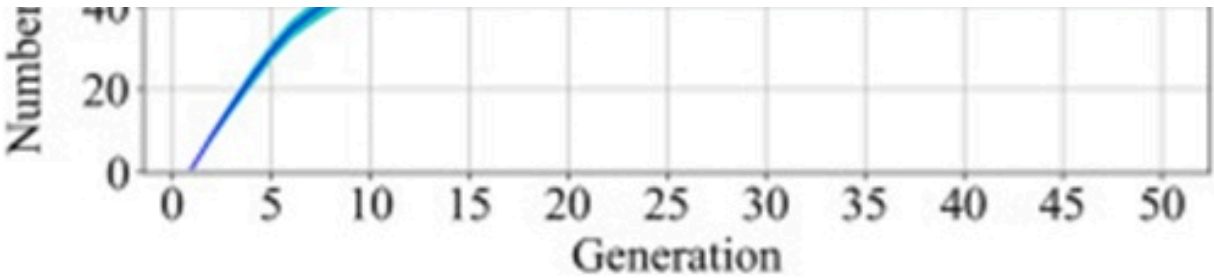
$M^{(g)}$: mean vector of parents selected in generation g

Figure 30 illustrates the optimization trajectory for Case A. At each step, the algorithm generates fresh candidates and steers the search using fitness feedback, steadily improving the quality of solutions. Because of the randomness inherent in CMA-ES, the full procedure was run 30 separate times; results are therefore expressed as mean ± one standard deviation. On average, the population means stabilized (Euclidean distance reached zero) by generation 22. The algorithm required evaluation of only 50 ± 7 distinct weightings from the 369 possibilities—far fewer than an exhaustive search would require. The final mean best fitness of 24.7 ± 2.2 indicates that, under asynchronous fan operation, the near-optimal weighting typically produces a total fan power very close to this fitness value.

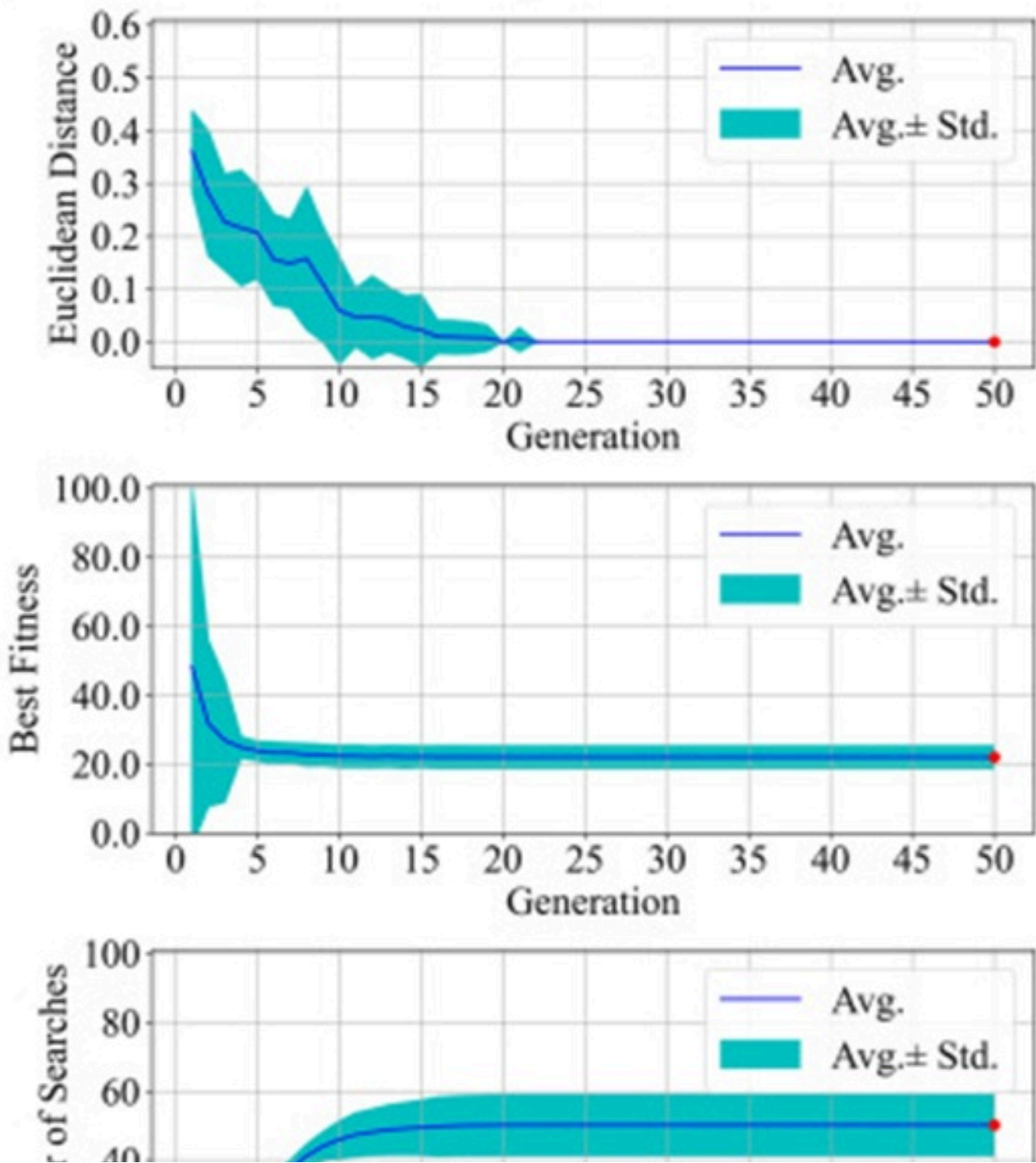
For Case B (**Figure 31**), the search process converged to a stable mean (Euclidean distance = 0) around generation 24. Across the 30 repetitions, an average of 52 ± 8 unique weightings were tested experimentally. The mean best fitness achieved was 25.2 ± 1.4 , implying that the discovered near-optimal weighting yields asynchronous fan power consumption roughly matching this fitness score.

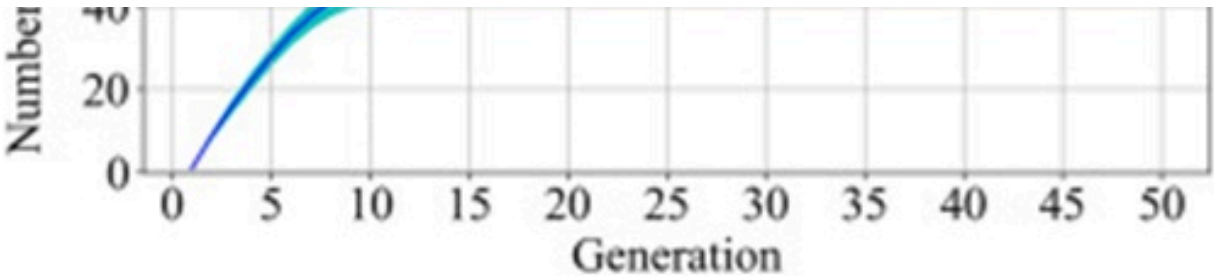
Weighting search results	Avg. ± Std.
Euclidean distance	0.0 ± 0.0
Best fitness	25.2 ± 1.4
Number of searches	52 ± 8



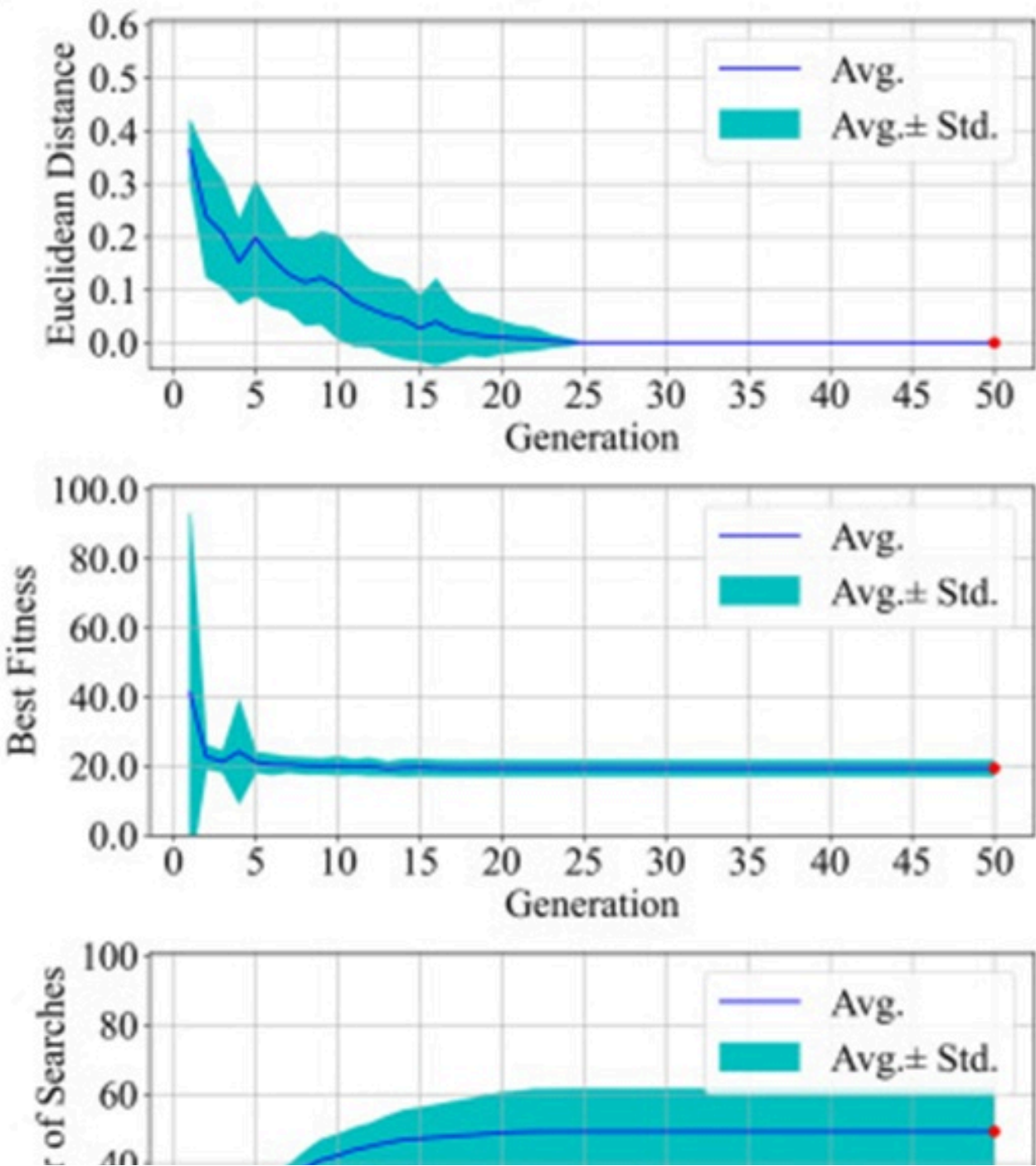


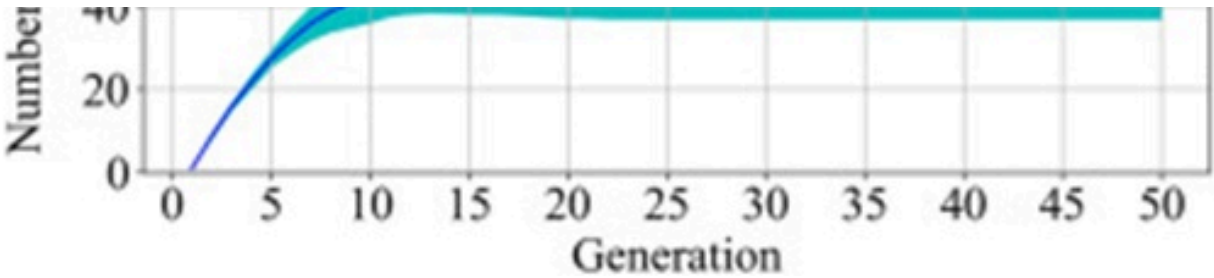
Weighting search results	Avg. ± Std.
Euclidean distance	0.0 ± 0.0
Best fitness	20.8 ± 2.9
Number of searches	50 ± 8



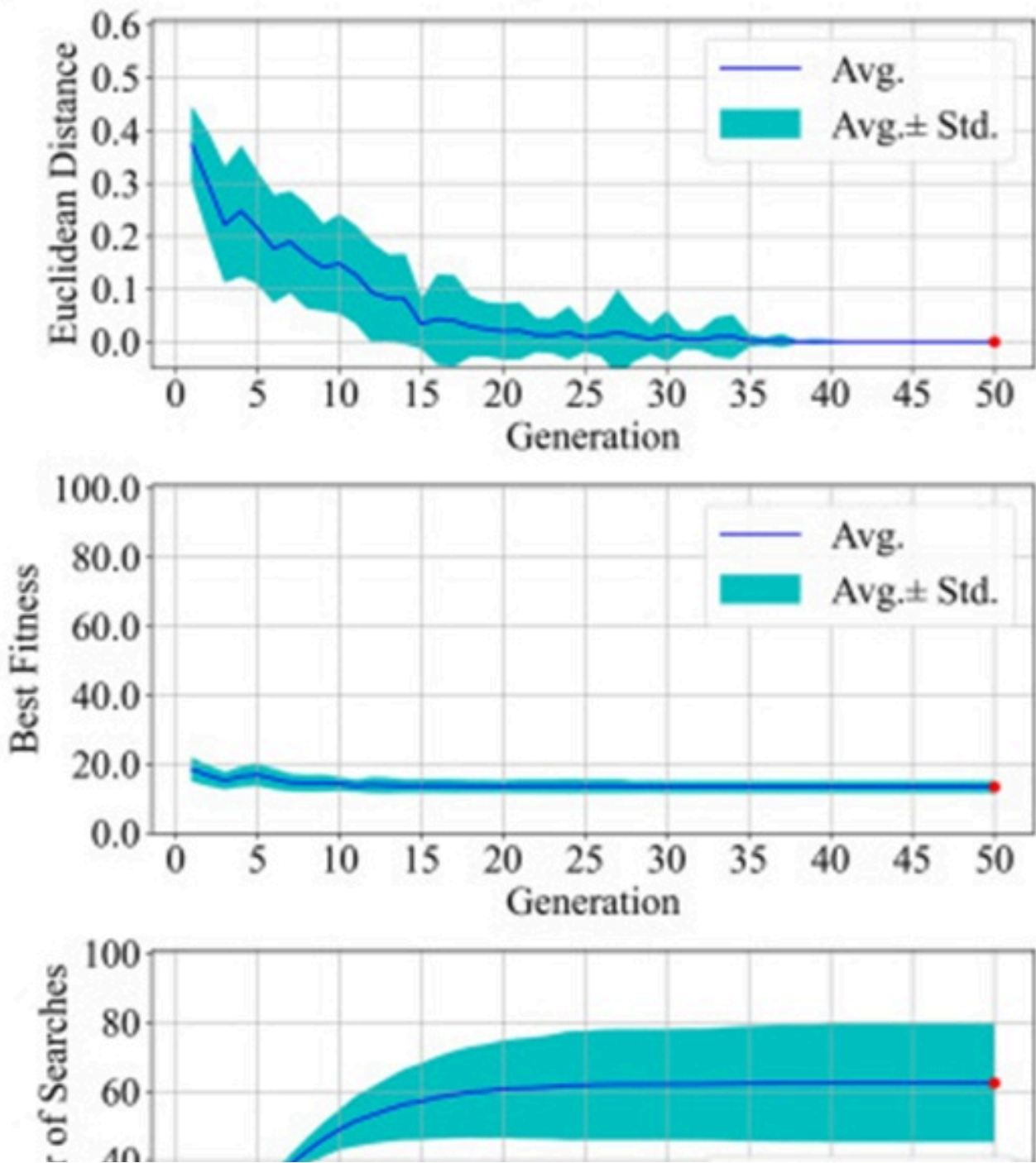


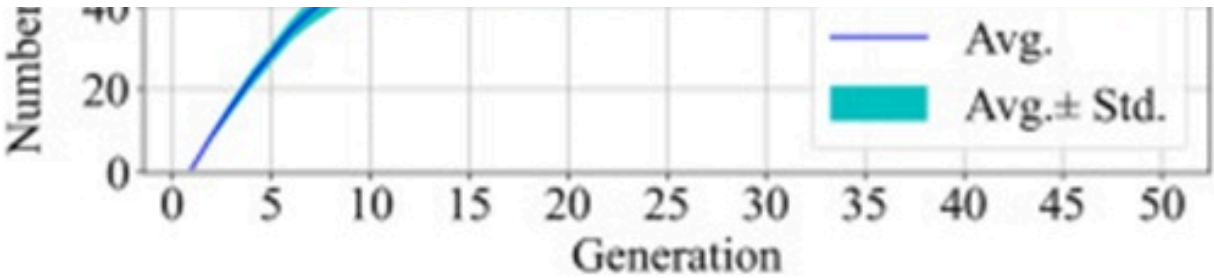
Weighting search results	Avg. ± Std.
Euclidean distance	0.0 ± 0.0
Best fitness	18.1 ± 1.5
Number of searches	49 ± 12



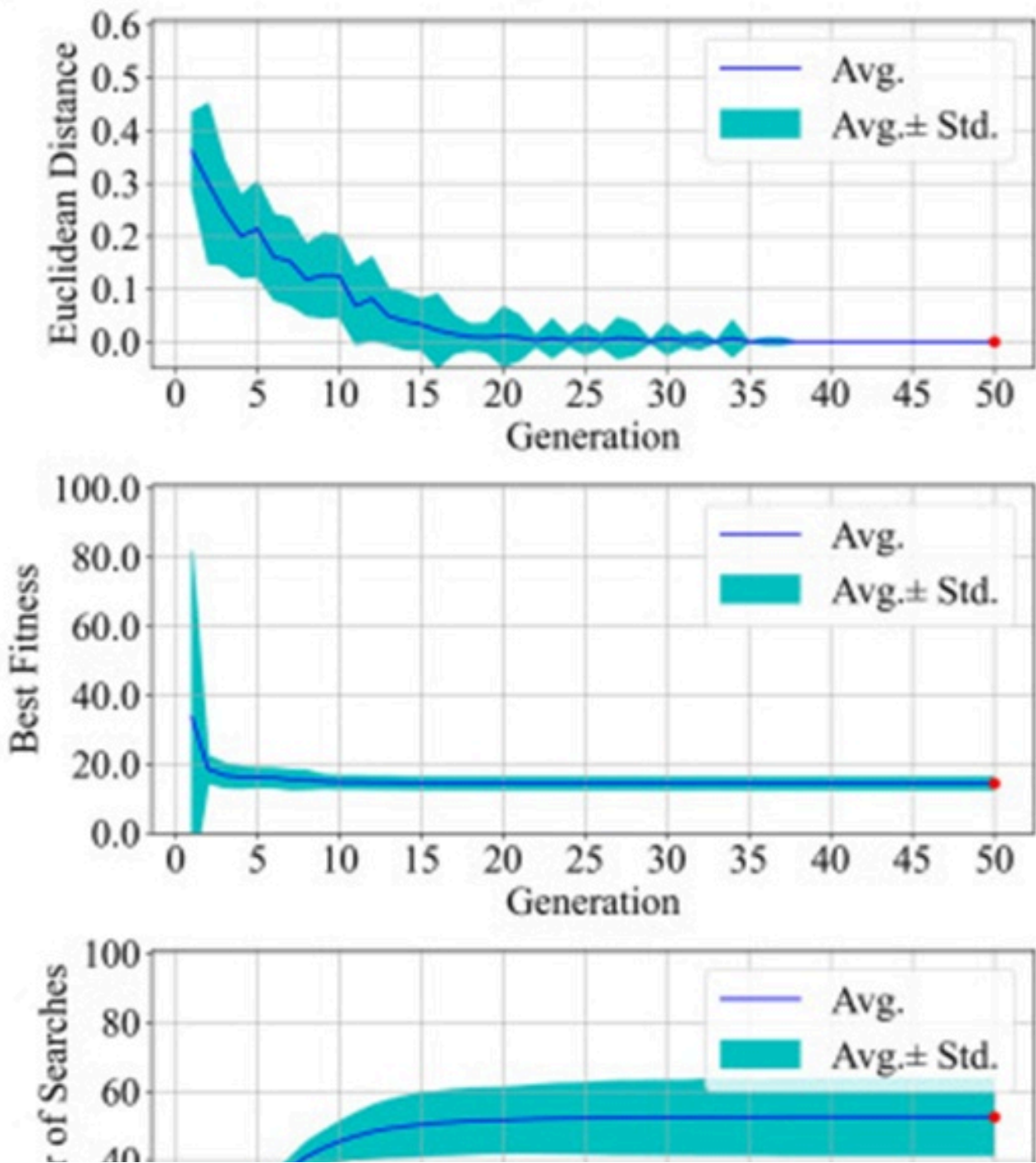


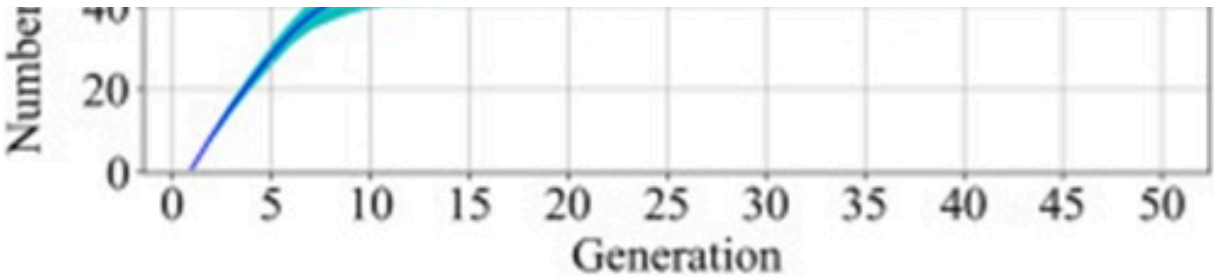
Weighting search results	Avg. ± Std.
Euclidean distance	0.0 ± 0.0
Best fitness	13.2 ± 1.1
Number of searches	62 ± 16



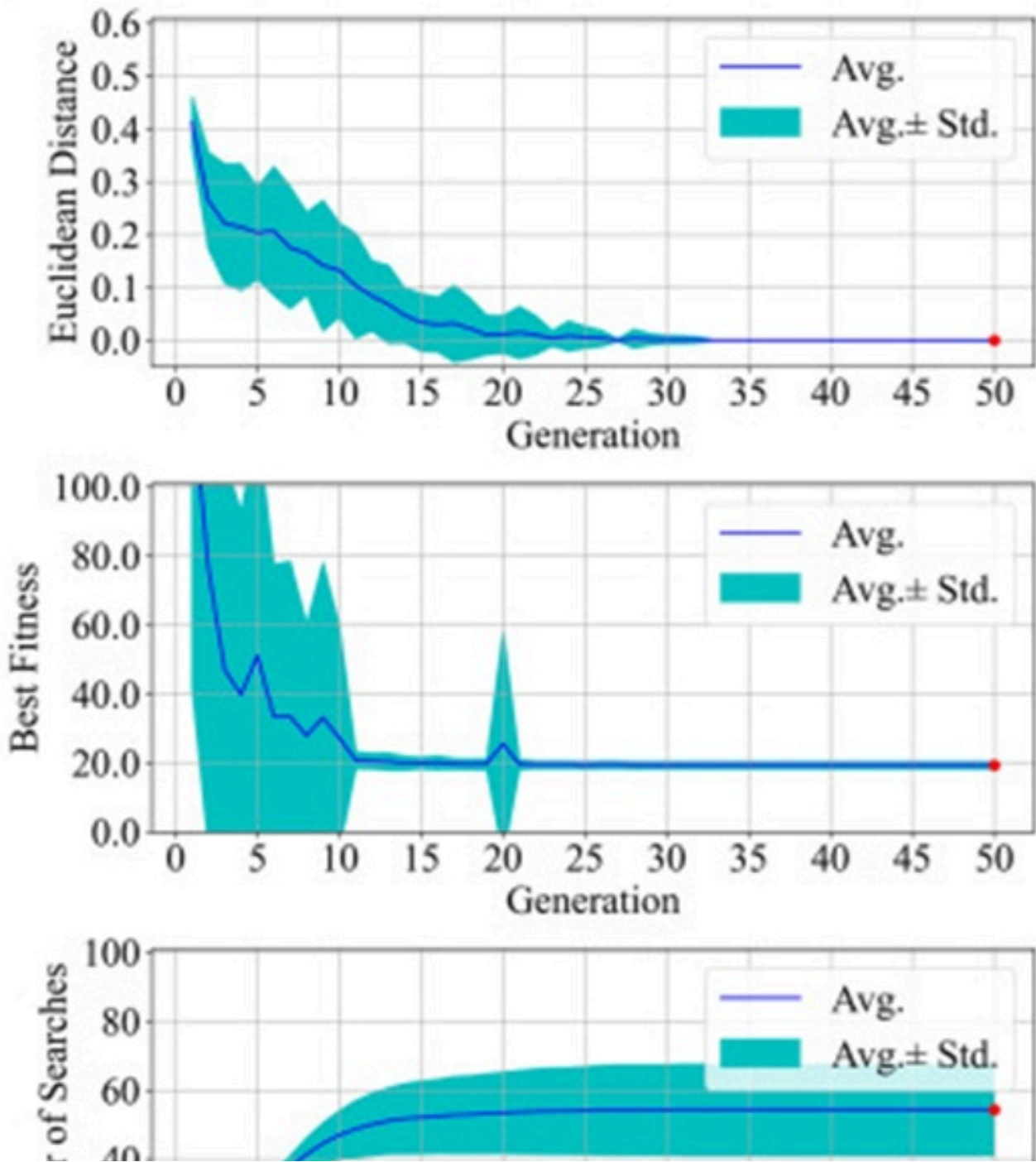


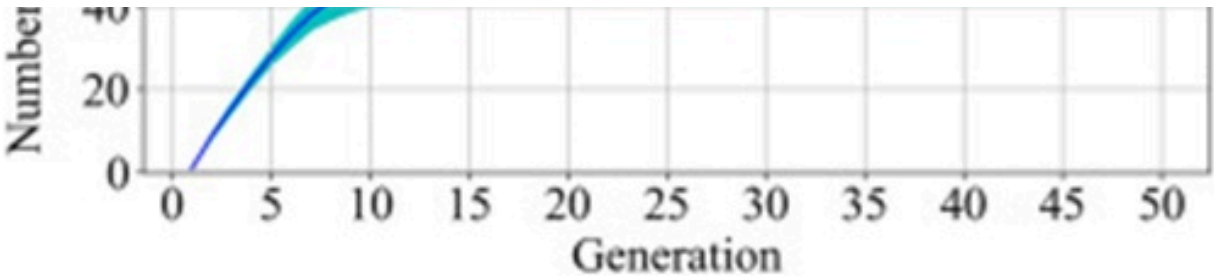
Weighting search results	Avg. ± Std.
Euclidean distance	0.0 ± 0.0
Best fitness	13.9 ± 1.1
Number of searches	52 ± 11



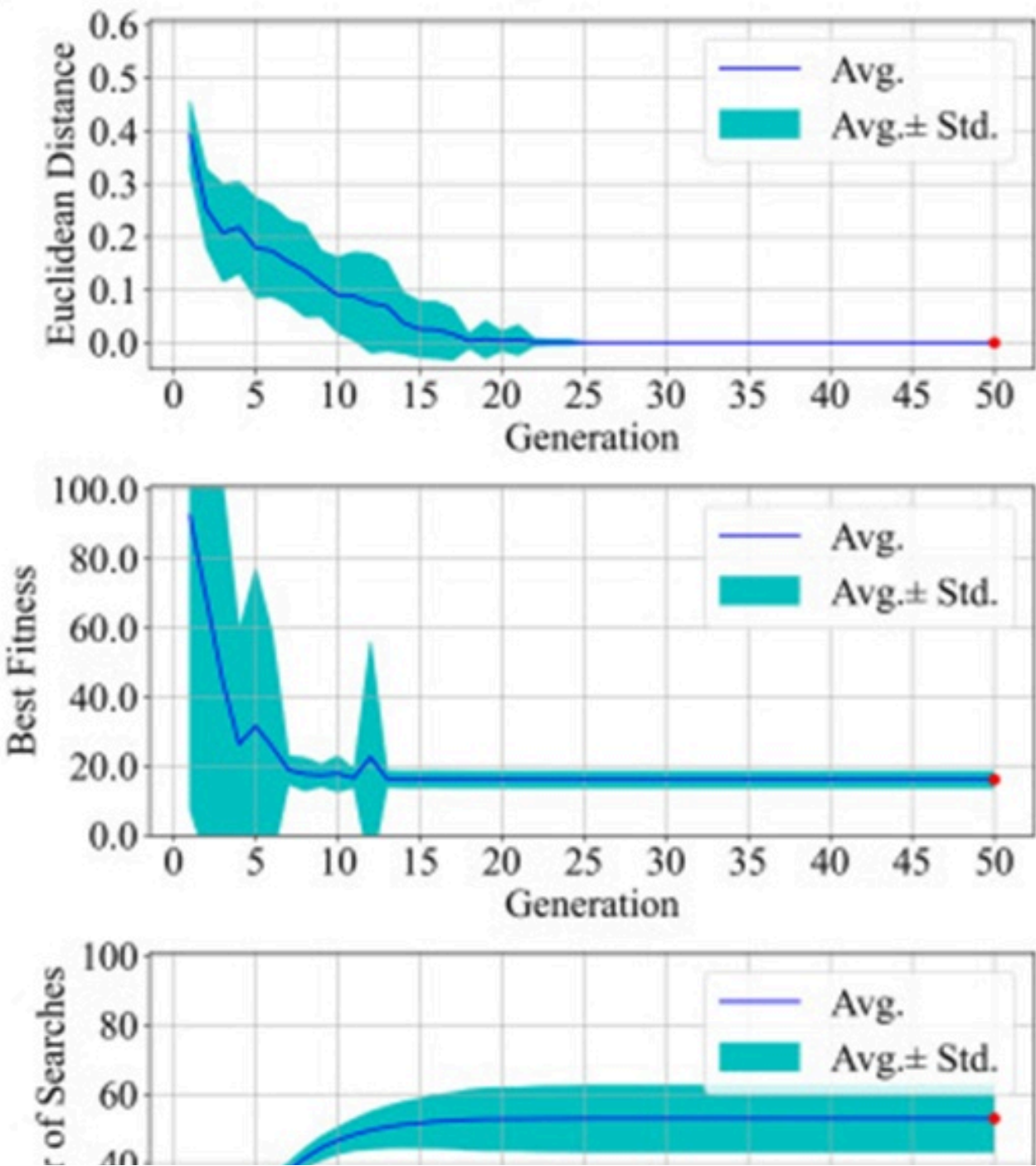


Weighting search results	Avg. ± Std.
Euclidean distance	0.0 ± 0.0
Best fitness	18.8 ± 0.5
Number of searches	54 ± 13





Weighting search results	Avg. ± Std.
Euclidean distance	0.0 ± 0.0
Best fitness	15.7 ± 1.8
Number of searches	52 ± 9



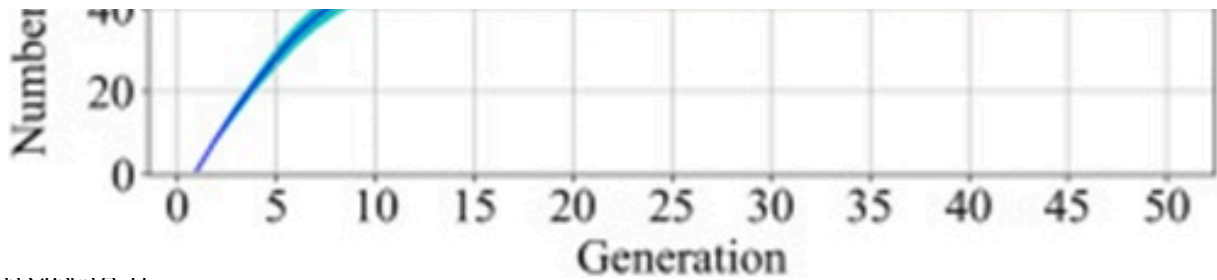


Figure 38: Results of the search-based approach.

When benchmarked against the full enumeration results from the weighting test phase, the search-based approach markedly raised the share of thermally acceptable weightings in Cases A–H by 15.2%, 21.7%, 20.6%, 18.9%, 22.7%, 18.2%, 14.4%, and 18.3%, respectively (Figure 38). These improvements highlight two key strengths: rapid convergence using relatively few evaluations and a simultaneous expansion of the set of practically usable weightings.

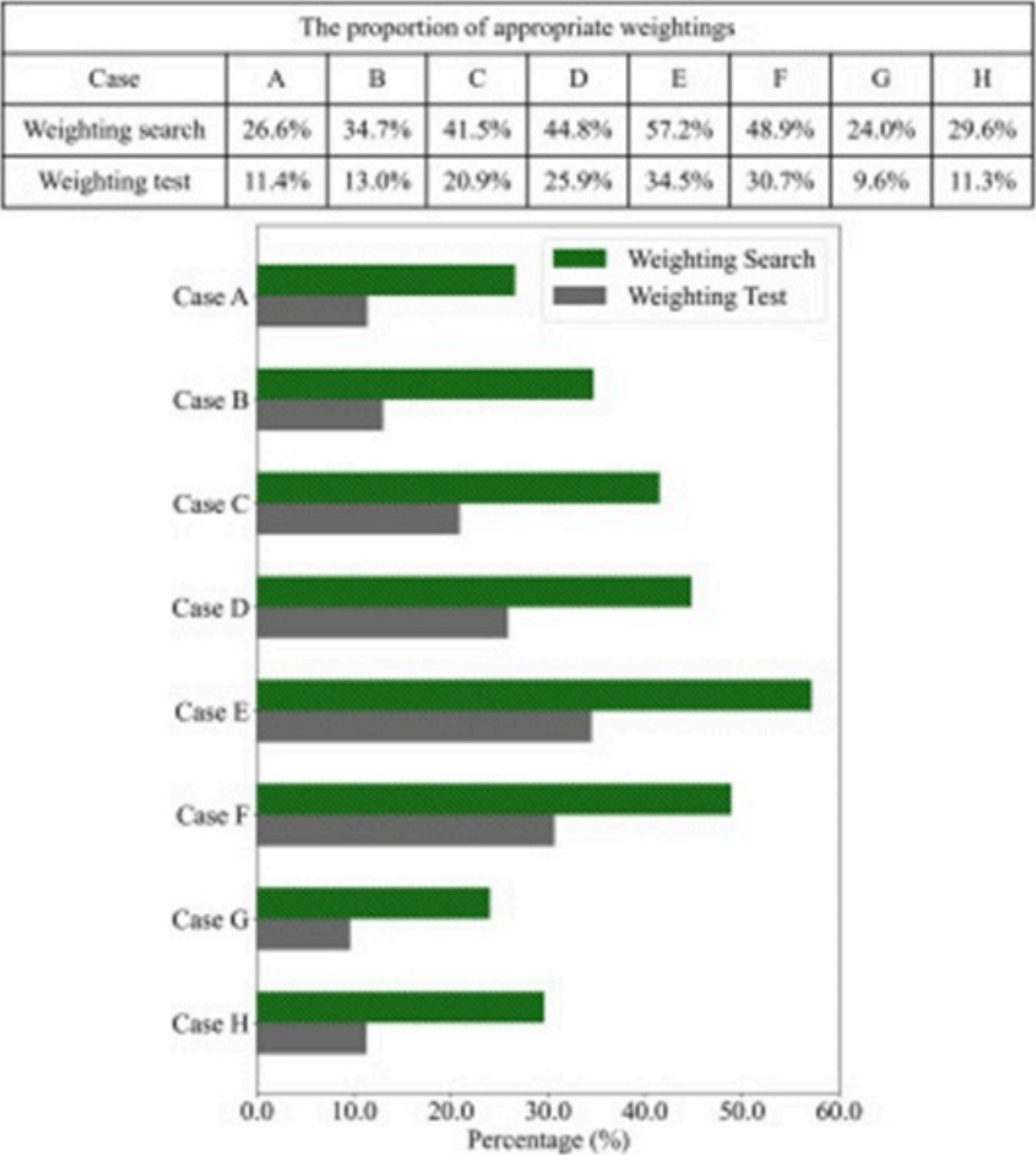


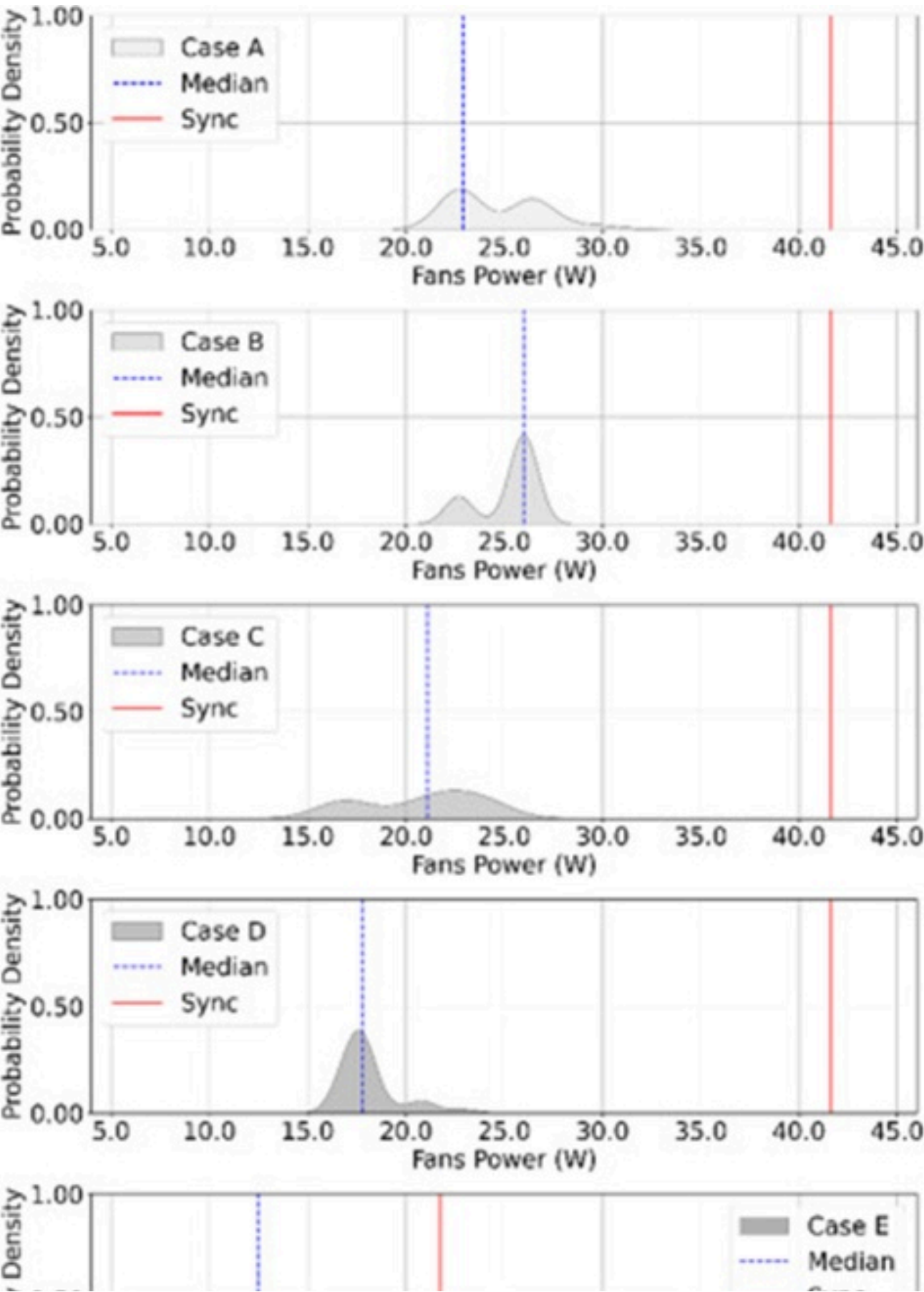
Figure 38. Weighting search and test — the proportion of appropriate weightings.

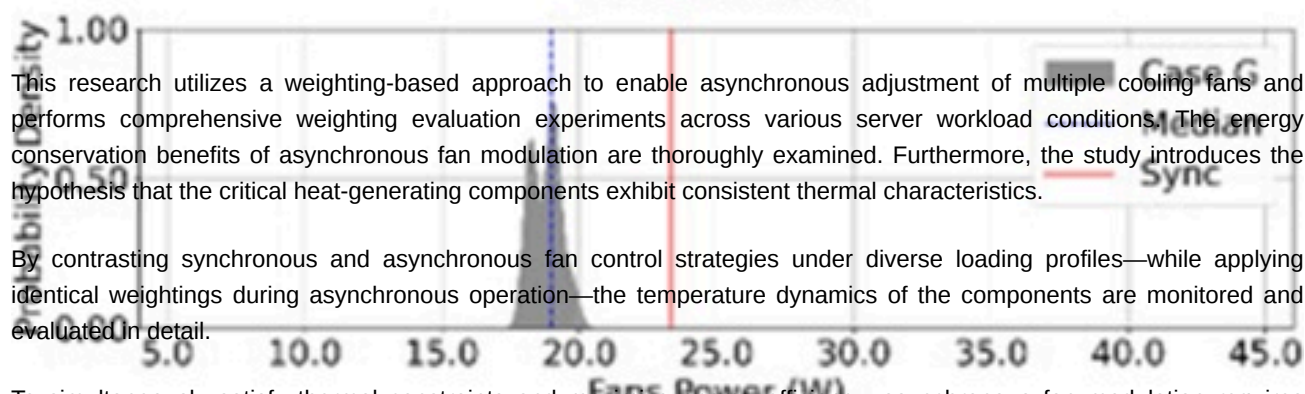
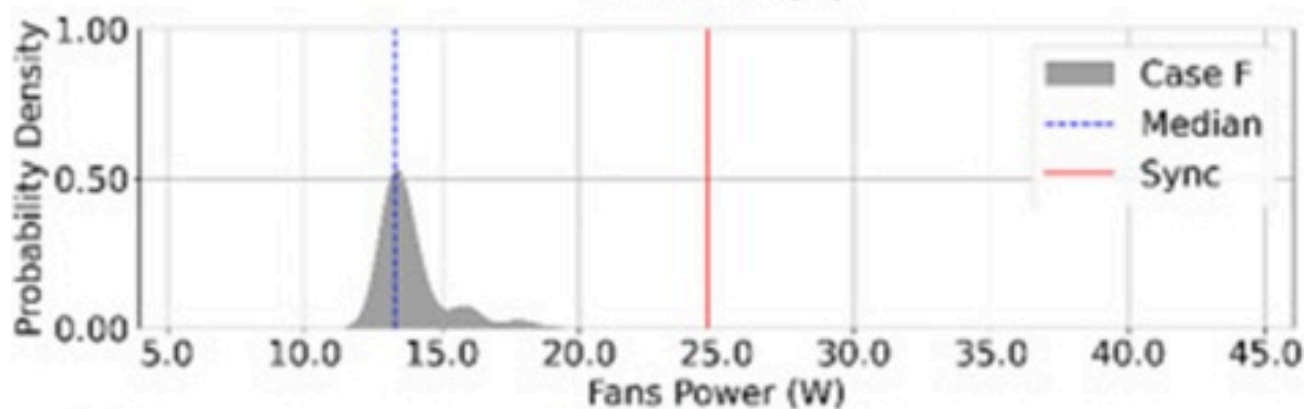
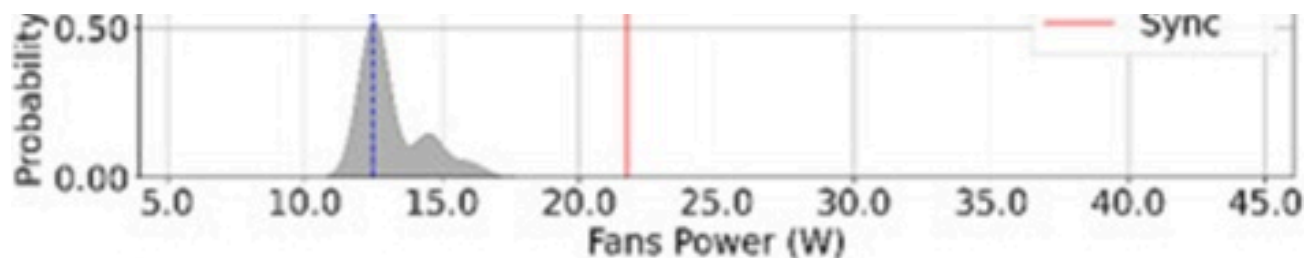
At the end of each search run, the best-fitness value obtained directly corresponds to the total fan power realized during asynchronous operation of the multi-fan control system. Table 9 provides a statistical overview of asynchronous fan power outcomes (minimum, median, maximum, and mean) along with the mean energy-saving gain relative to the synchronous baseline. Smoothed probability density functions were computed via Kernel Density Estimation (KDE) [27] and are presented in Figure 39. Here, the horizontal axis shows total fan power in watts; grayscale intensity (from light to dark) represents the estimated probability density of asynchronous power across Cases A–H; a blue dashed line marks

the median of the asynchronous distribution; and a solid red line shows the synchronous baseline power. The clear separation visible in **Figure 39** confirms consistent energy reduction across all eight workload conditions.

Table 9. Weighting search — the system's fans' power statistics.

Case	Asynchronous multi-fan control – fan power (W)				Average energy saving vs. synchronous (%)	Synchronous fan control – fan power (W)
	Minimum	Median	Maximum	Average		
A	22.7	22.9	30.1	24.7	40.7%	41.6
B	22.7	26.0	26.2	25.2	39.5%	41.6
C	16.9	21.1	24.8	20.8	50.0%	41.6
D	16.9	17.8	23.0	18.1	56.5%	41.6
E	12.5	12.5	16.0	13.2	39.3%	21.7
F	13.2	13.3	17.8	13.9	43.7%	24.7
G	18.2	19.0	19.8	18.8	19.5%	23.3
H	15.3	15.3	25.1	15.7	43.9%	28.0





This research utilizes a weighting-based approach to enable asynchronous adjustment of multiple cooling fans and performs comprehensive weighting evaluation experiments across various server workload conditions. The energy conservation benefits of asynchronous fan modulation are thoroughly examined. Furthermore, the study introduces the hypothesis that the critical heat-generating components exhibit consistent thermal characteristics.

By contrasting synchronous and asynchronous fan control strategies under diverse loading profiles—while applying identical weightings during asynchronous operation—the temperature dynamics of the components are monitored and evaluated in detail.

To simultaneously satisfy thermal constraints and maximize energy efficiency, asynchronous fan modulation requires carefully selected weightings to guide fan operation. To efficiently identify suitable weightings, this work combines the weighting mechanism with an Evolutionary Strategy algorithm. It develops two distinct fitness evaluation functions, thereby establishing a complete multi-fan control framework tailored for server applications. The effectiveness of this multi-fan control system is rigorously assessed through weighting optimization (search) experiments, each repeated 30 times to minimize the impact of algorithmic randomness. During the process, the algorithm iteratively refines the search direction according to fitness feedback, successfully locating near-optimal weightings with a relatively small number of evaluations. Asynchronous modulation with these weightings consistently meets thermal requirements while approaching minimal energy consumption. The stochastic nature of the algorithm helps avoid premature convergence to local optima, making it well-suited for parameter tuning in complex engineering optimization tasks.

The experimental platform consists of a 2U server equipped with four cooling fans and key heat sources, including CPU0, CPU1, PCIe0, and PCIe1. The results affirm three main contributions: (1) the practicality of asynchronous fan modulation, (2) the validity of the proposed hypothesis regarding uniform thermal behavior among critical components, and (3) the overall effectiveness of the developed multi-fan control system for server thermal management.

1. Across the eight distinct server loading scenarios, exhaustive testing of all 369 weighting combinations during asynchronous modulation yields varying counts and percentages of thermally compliant (appropriate) weightings, as well as different optimal weightings per case. These differences stem from the varying power dissipation levels of the components under each workload, which in turn influence component temperatures, fan duty cycles, and aggregate fan power demand. In Cases A–H, the identified best weightings represent global optima within the 369-candidate space for

their respective scenarios, satisfying thermal limits while minimizing total fan power. Compared with synchronous modulation, asynchronous operation with the best weighting reduces the peak duty cycle demanded from any fan, alleviates overcooling in certain components without violating thermal specifications, and achieves substantial reductions in total fan power—yielding energy savings of 45.5%, 45.5%, 59.4%, 59.4%, 42.5%, 46.6%, 22.1%, and 45.3% across Cases A–H, respectively. These outcomes validate both the effectiveness and the practical feasibility of asynchronous fan modulation.

2. After mapping the positions of the critical heat-generating components relative to the cooling fans, the hypothesis is advanced that CPUs and PCIe modules display equivalent thermal responses across different loading conditions. When synchronous and asynchronous modulation are compared using the same weightings for asynchronous control, steady-state temperature measurements confirm that the CPUs exhibit thermally consistent behavior. This consistency arises from the airflow path: incoming cold air first absorbs heat from the front-row CPUs before reaching the rear-row PCIe components. As a result, the thermal load on front-row CPUs directly impacts the temperatures of downstream PCIe elements. When identical weightings are applied in asynchronous modulation and front-row CPU loads remain unchanged across scenarios, the PCIe components likewise exhibit matching thermal characteristics. Experimental evidence strongly supports this hypothesis of uniform thermal phenomena. Moreover, these observations clarify why optimal weightings differ across workloads and reinforce the conclusion that server thermal regulation constitutes a nonlinear control challenge.

The weighting-optimization experiments combined the weighting mechanism with an Evolutionary Strategy (ES). They formulated two distinct fitness functions to construct an effective multi-fan control system tailored for servers. These experiments were run 30 times for each of the eight server workload conditions to mitigate the effects of algorithmic stochasticity. Across Cases A to H, an average of only 53 distinct weightings—equivalent to 14.2% of the total 369 candidates—underwent actual experimental evaluation. In all cases, the Euclidean distance converged to zero, confirming successful stabilization of the search process. Relative to the exhaustive weighting test results presented, the search-based approach increased the fraction of thermally compliant (appropriate) weightings by an average of 18.8%. These findings demonstrate that the algorithm achieves rapid convergence using a modest number of evaluations while simultaneously expanding the pool of viable weightings.

The near-optimal weightings identified by the algorithm matched one of the global optima with an average probability of 47.5%. Here, the global optimum is defined as the weighting that minimizes total fan power under asynchronous modulation within the multi-fan control framework. In contrast, the least favorable near-optimal solutions—those associated with the highest total fan power during asynchronous operation—occurred at an average rate of just 4.2%. Even these suboptimal weightings nonetheless delivered energy savings compared with the synchronous baseline. Overall, across the eight loading scenarios, the multi-fan control system developed in this study reduced total fan power by an average of 43.1% relative to synchronous modulation. These outcomes strongly validate the superior performance of the proposed multi-fan control approach for server thermal and energy management.

The evolutionary strategy-based optimization method presented here is intended for offline execution: optimal weighting configurations are precomputed and then integrated into the Baseboard Management Controller (BMC) firmware before server deployment. This design eliminates any real-time computational burden during normal operation, as the fan control logic applies the appropriate pre-optimized weights according to the detected workload profile.

Nevertheless, additional research is needed to develop adaptive real-time optimization strategies that dynamically update weight values based on live sensor data. Future investigations could explore lightweight heuristic methods or reinforcement learning techniques to enable responsive on-the-fly adjustments while preserving low computational demands. Although the asynchronous fan control strategy proposed in this work has proven effective at optimizing energy efficiency per server, its performance in a full data center setting has not yet been evaluated. In particular, future studies should examine the effects of varying supply air pressures in containment-based cooling systems and the complex interplay between individual server fans and rack-level airflow patterns. Moreover, practical implementation of

asynchronous fan control in production server hardware will require integration through standardized interfaces such as OpenBMC. Subsequent efforts will concentrate on transitioning the current experimental framework into a production-ready solution that aligns seamlessly with established server management ecosystems. These topics represent critical avenues for continued research, broadening the scope and real-world applicability of the proposed control methodology beyond the laboratory environment.

Acknowledgements

The Inventec Corporation of Taiwan supported this study. The financial aid and the equipment provided by Inventec are gratefully acknowledged.

Financial support

None

Conflict of interest

None

Ethics statement

None

Received: 29 Sep 2022

Revised: 05 Nov 2022

Accepted: 15 Dec 2022

Published online: 20 March 2023

Rights and permissions

Open Access The author(s) retain copyright. This article is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License. It may be shared and adapted for non-commercial purposes with appropriate attribution, an indication of changes, and distribution of adaptations under the same license. Third-party material may be subject to separate terms identified in its credit line. View the license at <https://creativecommons.org/licenses/by-nc-sa/4.0/>.

References

- SR Group. Cloud service provider global market revenue trend. 2022. [Online]. <https://www.srgresearch.com/articles/q2-cloud-market-grows-by-29-despite-strong-currency-headwinds-amazon-increases-its-share>, accessed: 2022-10-01.
- SR Group. Hyperscale data center. 2022. [Online]. <https://www.srgresearch.com/articles/pipeline-of-over-300-new-hyperscale-data-centers-drives-healthy-growth-forecasts>, accessed: 2022-10-02.
- Hintemann R, Hinterholzer S. Energy consumption of data centers worldwide - how will the internet be-come green?, in Proc.: ICT4S, Lappeenranta, Finland, 2019.
- Wei X, Xiao J, Wang Z, Wang Z, Tian Y. Integration and development of energy and information network in the pan-Arctic region. *Glob Energy Interconnect*. 2019;2(6):504–51.

AKCP. The electricity consumption of data centers. 2022. URL <https://www.akcp.com/blog/the-real-amount-of-energy-a-data-center-use/>

Koot M, Wijnhoven F. Usage impact on data center electricity needs: A system dynamic forecasting model, *Appl Energy*. 2021;291:116798.

Paris Agreement. Report of the Conference of the Parties to the United Nations Framework Convention on Climate Change, 21st Session, Paris; 2015.

Lennan M, Morgera E. The Glasgow climate conference (COP26). *Int J Mar Coast Law*. 2022;37(1):137–51.

Grid T. Green grid metrics: describing datacenter power efficiency, [Online]. http://www.thegreengrid.org/gg_content/Green_Grid_Metrics_WP.pdf (2007).

Zhang Q, Meng Z, Hong X, Zhan Y, Liu J, Dong J, et al. A survey on data center cooling systems: technology, power consumption modeling and control strategy optimization. *J Syst Archit*. 2021;119:102253.

Wang Z, Bash C, Tolia N, Marwah M, Zhu X, Ranganathan P. Optimal fan speed control for thermal management of servers. In: *International Electronic Packaging Technical Conference and Exhibition*; San Francisco, CA, USA; 2009. p. 709–19.

Shin D, Kim J, Chang N, Choi J, Chung SW, Chung EY. Energy-optimal dynamic thermal management for green computing. In: *IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*; San Jo-se, CA, USA; 2009. p. 652–7.

Lee C, Chen R. Optimal self-tuning PID controller based on low power consumption for a server fan cooling system. *Sensors*. 2015;15(5):11685–700.

Acun B. Mitigating variability in HPC systems and applications for performance and power efficiency, Ph.D. thesis, University of Illinois at Urbana-Champaign (2017).

Brännvall R, Sarkinen J, Svartholm J, Gustafsson J, Summers J. Digital twin for tuning of server fan controllers. In: *2019 IEEE 17th International Conference on Industrial Informatics (INDIN)*. IEEE; 2019.

Sarkinen J, Brännvall R, Gustafsson J, Summers J. Experimental analysis of server fan control strategies for improved data center air-based thermal management. In: *2020 19th IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (ITherm)*. IEEE; 2020.

Chu WX, Lien YH, Huang KR, Wang CC. Energy saving of fans in air-cooled server via deep reinforcement learning algorithm. *Energy Rep*. 2021;7:3437–48.

Chu H-C, Lee C-M, Tung K-Y, Chen R. Time series prediction for thermal management in server applied with deep learning. In: *2022 25th International Conference on Mechatronics Technology (ICMT)*. IEEE; 2022.

Jonsson H, Moshfegh B. Modeling of the thermal and hydraulic performance of plate fin, strip fin, and pin fin heat sinks—Influence of flow bypass. *IEEE Trans Compon Packag Technol*. 2001;24(2):142–9.

Byquist TA, Crocker MT. Method and system of advance fan speed control. US Patent US 7,490,479 B2; 2009.

Eiben AE, Smith JE. Introduction to evolutionary computing. Springer; 2003;53.

Darwin C. On the origin of species, 1859, Routledge, 2004.

Kern S, Müller SD, Hansen N, Büche D, Ocenasek J, Koumoutsakos P. Learning probability distributions in continuous evolutionary algorithms—a comparative review. *Nat Comput*. 2004;3(1):77–112.

Schwefel H-P. Numerical optimization of computer models. New York: John Wiley and Sons; 1981.

Hansen N, Ostermeier A. Completely derandomized self-adaptation in evolution strategies. *Evol Comput*. 2001;9(2):159–95.

Hansen N. The cma evolution strategy: a tutorial, arXiv preprint arXiv:1604.00772 (2016).

Lunacek M, Whitley D, Sutton A. The impact of global structure on search. In: *Parallel Problem Solving from Nature (PPSN X): 10th International Conference; Dortmund, Germany*. Berlin: Springer; 2008. p. 498–507.