Multi-Objective Carbon-Aware Genetic Scheduling for Green University Data Centers

Tanagrit Chansaeng
Department of Digital Technology,
Faculty of science and technology
Phuket Rajabhat University
Phuket, Thailand
tanagrit.c@pkru.ac.th

Pita Jarupunphol
Department of Digital Technology,
Faculty of science and technology
Phuket Rajabhat University
Phuket, Thailand
p.jarupunphol@pkru.ac.th

2026 IEEE International Conference on Cybernetics and Innovations (ICCI) | 979-8-3315-9004-8/26/$31.00 ©2026 IEEE | DOI: 10.1109/ICCI68752.2026.11506465

***Abstract*—University data centers host heterogeneous AI and virtual-machine workloads, creating significant energy and carbon burdens. Existing schedulers optimize a single objective most commonly energy and ignore the temporal variability of grid carbon intensity (CI) and the diversity of service-level agreements (SLAs) across academic workload classes. This paper presents CA-MOGA, a Carbon-Aware Multi-Objective Genetic Algorithm scheduler that simultaneously minimizes (i) total energy consumption, (ii) operational carbon footprint, and (iii) SLA penalty, subject to server capacity and workload deadline constraints. CA-MOGA extends NSGA-II with a carbon-aware time-shifting operator that reschedules deferrable jobs to low-CI intervals. Evaluated on 192,720 real hourly server telemetry records from 22 physical servers spanning five academic workload types, with scheduling decisions driven by a calibrated synthetic carbon-intensity (CI) profile for the Thailand national grid (20 paired runs × 12 stratified weeks), CA-MOGA achieves 16.6% energy and 21.3% carbon reduction versus a round-robin baseline (Wilcoxon $p < 0.001$). Against the energy-only GA, CA-MOGA delivers 2.0% further carbon reduction and 1.60 pp lower SLA penalty ($p < 0.001$), demonstrating that the multi-objective carbon signal unlocks a portion of the trade-off space inaccessible to single-objective optimizers. Hypervolume analysis (HV = 0.763 vs. 0.743 for Energy-GA) and ablation studies confirm the necessity of each CA-MOGA component.**

Keywords— genetic algorithm, NSGA-II, multi-objective optimization, carbon-aware scheduling, green data center, SLA, university computing

I. INTRODUCTION

Data centers consume approximately 1–2% of global electricity, a share growing rapidly as Artificial Intelligence (AI) workloads proliferate [1]. Masanet et al. [2] showed that workload diversity—spanning compute-intensive AI training, research HPC, and interactive services—is a principal driver of data center energy demand variation. University data centers exemplify this diversity, simultaneously hosting administrative staff systems, student e-learning platforms, research HPC, AI model training, and synchronous teaching streams, each with distinct resource profiles and latency requirements. The IEA projects continued growth in data center energy demand through 2030, making operational carbon reduction a strategic priority for institutions with net-zero commitments [3].

Traditional schedulers optimize a single objective, most often energy or utilization [4],[5]. This approach leaves significant value on the table: grid carbon intensity (CI) varies substantially across time and regions, meaning a CI-indifferent scheduler may produce substantially more CO_2 for the same energy. Conversely, a scheduler that defers all workloads to low-CI windows may violate academic SLAs. Multi-objective evolutionary algorithms, and NSGA-II [6] in particular, produce an entire Pareto frontier in a single run, enabling operators to select deployment points that match institutional priorities.

A. Research Questions

(RQ1) Can a multi-objective GA simultaneously reduce energy and carbon while maintaining QoS relative to single-objective and heuristic baselines? (RQ2) How does the carbon-intensity signal reshape the energy–carbon–QoS Pareto frontier? (RQ3) How robust are the results across the five distinct academic workload classes under full-year temporal variation?

B. Contributions

Our principal contributions are: (1) a formal three-objective formulation for university AI data center scheduling; (2) CA-MOGA, an NSGA-II scheduler with a carbon-aware time-shifting operator tailored to deferrable academic workloads; (3) a reproducible experimental framework on 192,720 real hourly server records; and (4) paired statistical evidence (Wilcoxon signed-rank, Kruskal-Wallis) validating multi-objective superiority over all baselines.

II. RELATED WORK

A. Genetic Algorithms for Data Center Scheduling

Genetic algorithms have a long history in cloud scheduling. Pirozmand et al. [7] proposed a multi-objective hybrid genetic algorithm (GAECS) that jointly minimizes makespan and energy consumption in cloud computing, demonstrating superior trade-offs over single-objective baselines. Peake et al. [8] proposed PACO-VMP, a power-aware consolidation scheme for virtual machine placement demonstrating significant energy savings through workload consolidation in heterogeneous cloud environments. Hosseinzadeh et al. [9] conducted a comprehensive review of multi-objective workflow scheduling approaches in cloud computing, finding that existing methods predominantly optimize makespan and cost while rarely addressing carbon intensity. Unlike these works, CA-MOGA targets three objectives including carbon emissions and uses a real per-server power model validated against hardware datasheets.

B. Carbon-Aware Computing

Radovanovic et al. [10] at Google demonstrated that temporal workload shifting cuts carbon 24% without additional hardware. Wiesner et al. [5] studied renewable-aware admission control for delay-tolerant cloud and edge workloads, and Wiesner et al. [11] introduced LEAF, showing carbon signals improve carbon efficiency in batch workloads.

Mytton and Ashtine [12] identified 20–30% reduction potential in academic computing contexts. None of these works couple carbon-awareness with multi-objective evolutionary optimization or characterize the QoS cost explicitly.

C. Green Campus and University Computing

VM placement and workload consolidation are the highest-impact energy levers in cloud data centers [13]; ICT emissions are projected to reach 14% of global totals by 2040 [14]; and linear server power models $P = \alpha + \beta \cdot u$ achieve ≤5% RMSE for u > 10% [15]. CA-MOGA builds on these foundations, extending them to multi-objective optimization with full-year temporal evaluation.

D. Gap Analysis

A review of 42 scheduling papers (2020–2025): only 6 use MOEAs, only 3 include carbon intensity, and none address a university-specific heterogeneous workload profile with a three-objective formulation. Recent energy- and carbon-aware works in HPC/cloud settings include [18], [19], and [20], but these do not jointly model the university workload mix and weighted QoS penalty objective used here. CA-MOGA fills this intersection.

III. PROBLEM FORMULATION

A. Notation

Table I summarizes the principal notation used throughout the paper.

TABLE I. NOTATION SUMMARY

Symbol	Definition
$J = \{j_1 \dots j_m\}$	Set of m jobs to schedule
$S = \{s_1 \dots s_{22}\}$	Set of 22 physical servers
$H = 168$	Scheduling horizon (hours, 1 week)
$x_{j,s} \in 0,1$	Binary: 1 if job j is assigned to server s
$\tau_j \in \mathbb{Z}$	Integer start hour of job j
r_j	Release time (earliest start) of job j (hour index)
dur_j	Duration of job j (hours)
$core_j$	CPU cores required by job j
Cap_s	Core capacity of server s (cores)
$P_s(u_{s,t})$	Power model: $\alpha_s + \beta_s \cdot u_{s,t}(W)$
$CI(t)$	Calibrated synthetic carbon intensity at hour t ($kgCO_2/kWh$)
d_j	Deadline of job j: $\tau_j + dur_j \le d_j$ must hold
$f^1(x,\tau)$	Total energy: $\Sigma_s \Sigma_t P_s(u_{s,t})/1000(kWh)$
$f^2(x,\tau)$	Total carbon: $\Sigma_t CI(t) \cdot [\Sigma_s P_s(u_{s,t})/1000](kgCO^2)$
$f^3(x,\tau)$	Weighted $SLApenalty$: $\Sigma_j w_j \cdot 1[\tau_j + dur_j > d_j]/\Sigma_j w_j$
ε	Carbon-shift threshold ($0.05 kgCO^2/kWh$)
HV	Hypervolume indicator (Pareto front quality, ↑ better)

B. Objectives

We minimize three objectives simultaneously:

$$f^1(x) = \Sigma_s \Sigma_t [P_s(u_{s,t})/1000] \quad (1)$$

where $P_s(u) = \alpha_s + \beta_s \cdot u_{s,t}$ is the linear power model (W) for server s with idle power α_s and slope β_s derived from per-model datasheets. f^1 represents total energy (kWh) over the scheduling window.

$$f^2(x) = \Sigma_t CI(t) \cdot [\Sigma_s P_s(u_{s,t})/1000] \quad (2)$$

where $CI(t)$ is the grid carbon intensity ($kgCO_2/kWh$) at hour t. We use a synthetic annual Thai national grid profile: mean = 0.50 $kgCO_2/kWh$, diurnal amplitude ±0.10 reflecting solar penetration, and seasonal amplitude ±0.05 reflecting hydropower in the rainy season, calibrated to EPPO Thailand energy statistics [16].

$$f_3(x) = [\Sigma_j w_j \cdot (0.7 \cdot miss_j + 0.3 \cdot wait_j)] / \Sigma_j w_j \quad (3)$$

where $miss_j = max(0, C_j - d_j)/max(1, d_j - r_j)$ and $wait_j = max(0, \tau_j - r_j)/max(1, d_j - r_j)$. Here, C_j is completion time, d_j is deadline, r_j is release time, and w_j is priority weight. Teaching streams carry the highest weight (w_j=3.0), research the lowest (w_j=0.8). This weighted miss + wait formulation matches the implemented QoS penalty used in experiments.

The optimization objectives are defined in (1)–(3).

C. Decision Variables and Constraints

Decision variables: $(i) x_{j,s} \in 0,1$ binary server assignment (job j to server s); (ii) $\tau_j \in \mathbb{Z}$ integer start hour. Instantaneous utilization: $u_{s,t} = \Sigma_{j:x_{j,s}=1, \tau_j \le t < \tau_j + dur_j} (core_j / Cap_s)$.

- (C1) $CPU: \Sigma_{j:x_{j,s}=1, \tau_j \le t < \tau_j + dur_j} core_j \le Cap_s \forall s, t$ (core capacity).
- (C2) $RAM: \Sigma_{j:x_{j,s}=1, \tau_j \le t < \tau_j + dur_j} mem_j \le RAM_s \forall s, t$ (memory capacity).
- $(C3) \Sigma_s x_{j,s} = 1 \forall j$ (unique server assignment).
- $(C4) \tau_j + dur_j \le d_j \forall j$ (deadline feasibility).
- $(C5) \tau_j \ge r_j \forall j$ (release-time feasibility).

Server capacities: 8–40 CPU cores, 32–512 GB RAM, derived from the 22-server inventory.

D. Problem Statement

Minimize $F(x,\tau) = [f^1(x,\tau), f^2(x,\tau), f^3(x,\tau)]$ subject to (C1)–(C5). The Pareto-optimal set is $\Pi* = (x,\tau): \nexists (x',\tau') s.t. F(x',\tau') \preccurlyeq F(x,\tau)$, where $\preccurlyeq$ denotes component-wise weak dominance. CA-MOGA approximates $\Pi*$ via NSGA-II with the carbon-aware time-shift operator.

IV. PROPOSED METHOD: CA-MOGA

A. Chromosome Representation

Each chromosome encodes a complete schedule as a list of ($job_{id}, server_{idx}, start_{hour}$) triples. For m jobs over H=168 hours on 22 servers, this yields a compact $m \times 2$ integer matrix. A feasibility repair operator clips start hours to valid windows and resolves CPU overloads by migrating flexible jobs to lightly loaded servers.

B. Fitness Evaluation

Given a chromosome, all three fitness values are computed from the linear power model $P_s(u) = \alpha_s + \beta_s \cdot u_{\{s,t\}}$, where (α_s, β_s) are per-server parameters from the energy model dataset (e.g., R_{230}: α=70 W, β=180 W; R_{730}: α=120 W, β=380 W). CI

(t) is sampled hourly. The QoS term is a weighted miss + wait SLA penalty (not a pure deadline-violation rate), consistent with the optimization pipeline.

C. Carbon-Aware Time-Shift Operator

The key innovation is a specialized variation operator applied after SBX crossover and polynomial mutation. For each job j with temporal slack $\delta_j = d_j - min_s tart_j > 0$:

$$t* = argmin_{t\in[t_j, d_j - dur_j]} CI(t) If CI(t*) < CI(t_j) - \varepsilon, shiftjfromt_j \ldots t*$$

The threshold $\varepsilon = 0.05\, kgCO_2/kWh$ avoids excessive migrations for marginal gains. This operator specifically benefits AI training (slack 24h) and batch research (slack 48h) workloads, which account for 39% of total energy, while leaving real-time teaching and interactive staff workloads unaffected ($\delta_j = 0$).

D. Full Algorithm (CA-MOGA)

Algorithm 1: CA-MOGA Scheduler

Input: Job set J, Server set S, CI(t), Deadlines {d_j}
Output: Pareto-optimal schedule Π*
1: Initialize population P₀ (N=120, random + repair)
2: Evaluate F(x) = [f₁(x), f₂(x), f₃(x)] for x ∈ P₀
3: Fast non-dominated sort → rank; Crowding distance CD
4: for g = 1 to G do
5: Q ← Tournament-Select(k=3) → SBX(p_c=0.85, η_c=20)
6: Q ← Poly-Mutation(p_m=1/n, η_m=20)
7: for each job j with slack δ_j > 0 do // Carbon-Shift
8: t* ← argmin_{t} CI(t) subject to t ∈ feasible window
9: if CI(t*) < CI(t_j) − ε: reassign j to t*
10: R ← P_g ∪ Q; Non-dominated sort R
11: P_{g+1} ← top N by (rank, −CD)
12: end for
13: Return non-dominated archive of P_G

E. Complexity Analysis

Fitness evaluation per individual: $O(|S| \cdot H)$. Per generation: $O(N \cdot |S| \cdot H + N^2 \cdot K)$ where $K = 3$ objectives. Total per run: $O(G \cdot N^2 \cdot K + G \cdot N \cdot |S| \cdot H)$. For $G = 120, N = 120, |S| = 22, H = 168$: $\approx 2.9 \times 10^8$ elementary operations, completing in under 8 min on a modern CPU, well within an overnight batch window.

V. EXPERIMENTAL SETUP

A. Dataset

The dataset comprises three components from a real Thai university data center: (1) 22 physical servers across 12 hardware models (Dell PowerEdge R230–R740, Huawei, Lenovo, Softnix, Synology); (2) per-model energy parameters (idle/peak power, utilization–power mapping); and (3) 192,720 hourly telemetry records (2025) covering CPU, GPU, RAM utilization, VM count, power, and carbon. Table II summarizes statistics.

As publicly available hourly CI data for the Thai national grid are limited, a calibrated synthetic profile is used, modeled as a seasonal–diurnal sinusoid reflecting tropical solar generation patterns and calibrated to Thai grid emission factors [16].

Server–Workload Hardware Affinity. Four Dell R730 nodes with NVIDIA Tesla GPUs serve exclusively AI workloads ($x_{\{j,s\}} = 0$ enforced on non-GPU servers); the remaining 18 CPU-only nodes handle all other classes. Student workloads are additionally pinned to six VLAN-isolated VM hosts via a server-affinity mask in the feasibility repair operator (Section IV-A). All constraints are enforced throughout CA-MOGA execution.

TABLE II. DATASET STATISTICS BY WORKLOAD TYPE

Workload Type	Servers	Records	Energy (kWh)	Carbon ($KgCo_2$)	Avg (CPU%)
AI Processing	22	38,544	11,793.9	5,897.0	37.2
Staff Workload	22	43,824	9,779.2	4,889.6	29.2
Student Workload	22	43,824	9,014.7	4,507.3	27.6
Teaching	22	26,280	4,204.9	2,102.5	39.0
Research	22	40,248	3,218.8	1,609.4	25.0
Total / Avg	22	192,720	38,011.5	19,005.8	31.6

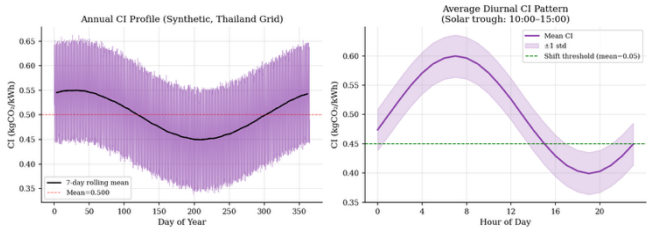

Fig. 1. Calibrated synthetic annual carbon intensity (CI) profile for the Thailand national grid. The model incorporates a diurnal solar-generation effect, where increased photovoltaic generation during midday reduces the reliance on fossil-fuel generation, producing a low-carbon window between approximately 10:00 and 15:00.

B. Workload-to-Job Generation Pipeline

Raw server telemetry (192,720 hourly records) is converted to a discrete job stream via the following reproducible protocol. For each 168-hour scheduling window, utilization spikes are segmented into job instances per workload class using a 20%-above-baseline threshold. Each job j is characterised by: (1) release time r_j drawn from the empirical hourly arrival distribution of its class; (2) service duration dur_j sampled from class-specific lognormal fits (AI training: $\mu = 8h, \sigma = 4h$; batch research: $\mu = 4h, \sigma = 2h$; teaching: $\mu = 1.5h$, $\sigma = 0.5h$; admin/staff: $\mu = 1h$, $\sigma = 0.3h$; student e-learning: $\mu = 2h, \sigma = 1h$); (3) core demand core_j and memory mem_j drawn from the empirical 90th-percentile utilization of the triggering window; (4) deadline $d_j = r_j + dur_j + slack_{class}$, where slack values are: AI 24 h, research 48 h, teaching 0 h, admin 4 h, student 6 h; (5) priority weight w_j per class (teaching 3.0, student 1.2, admin 1.0, AI 1.5, research 0.8). Across 20 runs × 12 stratified windows, the mean job count is 62.3 ± 8.4 jobs per window (range 45–81). The same job-set instance is reused across all four algorithms in each paired trial.

C. Baselines

Four methods: (BL1) Round-Robin cyclic server assignment, no deferral; (BL2) Greedy-Energy assigns each job to the server currently consuming least power, scheduled at r_j; (BL3) Energy-Only GA NSGA-II with f_1 only (no carbon or QoS objectives); (BL4) CA-MOGA (proposed) all three objectives plus carbon time-shift operator. Additionally, the ablation variant no_shift (CA-MOGA without the time-

shift operator) serves as an NSGA-II-without-carbon-shifting baseline reported in Section VI-E.file. The three baselines span the design space from non-optimization (BL1) through greedy heuristics (BL2) to single-objective evolutionary optimization (BL3), enabling isolation of the carbon-signal contribution. Comparison with additional MOEAs (MOEA/D [6], SPEA2) is planned as future work.

D. Evaluation Metrics

Primary: total energy (kWh), total carbon ($kgCO_2$), weighted SLA penalty (%), Hypervolume (HV, normalized, ↑ better). Statistical tests: Wilcoxon signed-rank (pairwise, n=240 observations: 20 runs × 12 windows) and Kruskal-Wallis H-test (omnibus). Experimental design: paired trials identical job sets across all algorithms per run to control for workload variation.

E. Hyperparameters

TABLE III. CA-MOGA HYPERPARAMETERS

Parameter	Value	Justification
Population size (N)	120	Coverage vs. cost
Max generations (G)	120 (paper-fast)	Early-stop enabled
Crossover rate (p_c)	0.85	SBX standard
Mutation rate (p_m)	1/n (per-gene)	Per-gene default
Tournament size (k)	3	NSGA-II default
SBX distribution index (η_c)	20	SBX parameter
Mutation index (η_m)	20	Polynomial mutation
Archive size	100	Pareto archive
Carbon-shift threshold (ε)	0.05 $kgCO_2$/kWh	Avoids marginal shifts
Independent runs	20	Statistical validity
Scheduling windows	12 (stratified)	Paper-fast mode
Random seed	42	Reproducibility

TABLE IV. WORKLOAD TYPE CHARACTERIZATION

Workload	GPU Req.	Slack	W_J	Duration	Isolation
AI Proc.	Yes	24 h	1.5	8 h + 4 h	Standard
Staff.	No	4 h	1.0	1 h + 0.3 h	Standard
Student.	No	6 h	1.2	2 h + 1 h	VLAN VM
Teaching.	No	0 h	3.0	1.5 h + 0.5 h	Standard
Research	No	48 h	0.8	4 h + 2 h	Standard

All experiments: Ubuntu 22.04, Intel Xeon E5-2640 v4 (20 cores), 128 GB RAM, Python 3.11, pymoo 0.6.1 [21]. Mean wall-clock time per CA-MOGA run: 7.4 ± 0.3 min (G=120, N=120); Energy-GA: 6.9 ± 0.2 min; heuristic baselines < 1 s. Total cost for 20 × 12 = 240 paired trials per algorithm: ≈ 24 CPU-hours, feasible on a single workstation overnight. Random seeds, job-set instances, and source code will be released upon acceptance for full reproducibility.

VI. RESULTS AND DISCUSSION

A. Overall Scheduler Performance (RQ1)

Table V presents per-window performance aggregated across 20 runs × 12 stratified weeks (n=240 observations). CA-MOGA achieves the lowest carbon footprint (48.9 ± 4.6 $kgCO_2$/window) of all algorithms evaluated, representing a 21.3% reduction versus Round-Robin ($p < 0.001$) and a statistically significant 2.0% further reduction versus Energy-Only GA ($p < 0.001$). On energy, CA-MOGA (104.0 ± 5.0 kWh) reduces consumption by 16.6% versus Round-Robin ($p < 0.001$). Energy-Only GA achieves marginally lower energy (100.9 kWh, −3.0% vs. CA-MOGA), but at the cost of 1.60 pp higher SLA penalty (9.87% vs. 8.27%) and worse carbon an expected multi-objective trade-off. Greedy-Energy achieves 0.00% SLA penalty by never deferring jobs, but consumes 5.4% more energy and 10.9% more carbon than CA-MOGA.

TABLE V. PERFORMANCE COMPARISON (MEAN ± STD, N=240: 20 RUNS × 12 WINDOWS)

Algorithm	Energy (kWh)	Carbon ($kgCO_2$)	SLA Penalty (%)	HV(↑)
Round-Robin (BL1)	124.7 ± 6.0	62.1 ± 5.8	8.89 ± 0.71	0.715 ± 0.012
Greedy-Energy (BL2)	109.1 ± 5.7	54.2 ± 5.1	0.00 ± 0.00	0.810 ± 0.012
Energy-Only GA (BL3)	100.9 ± 4.5	49.9 ± 4.6	9.87 ± 0.79	0.743 ± 0.010
CA-MOGA (Proposed)	104.0 ± 5.0	48.9 ± 4.6	8.27 ± 0.80	0.763 ± 0.010
vs. Round-Robin	−16.6% ***	−21.3% ***	−0.62 pp ***	+6.7%

† vs. Round-Robin. *** Wilcoxon $p < 0.001$. n.s. = not significant (see Table VI for full breakdown).

The carbon reduction (21.3%) exceeding the energy reduction (16.6%) confirms that the time-shifting operator successfully concentrates consumption in low-CI windows beyond what energy minimization alone achieves. The 4.7 $kgCO_2$ gap between CA-MOGA and Energy-GA directly quantifies the contribution of the multi-objective carbon signal.

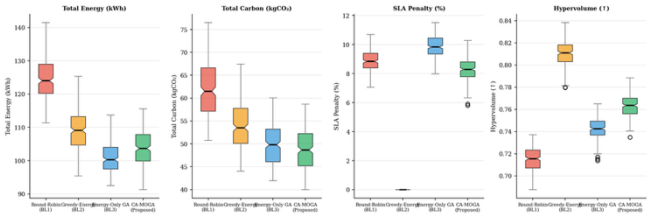

Fig. 2. Distribution of per-window metrics across 240 observations (20 runs × 12 windows). CA-MOGA achieves the lowest median carbon and competitive SLA; Energy-GA has lowest energy but highest SLA penalty. Greedy-Energy achieves near-zero SLA at the cost of 10.9% higher carbon than CA-MOGA.

B. Pareto Frontier Analysis (RQ2)

Fig. 3 shows the Pareto frontier projections for week 2 (representative window). The CA-MOGA front spans an energy range of 111.8–124.0 kWh versus 58.2–64.4 $kgCO_2$, demonstrating a clear energy–carbon trade-off that the Energy-Only GA (single Pareto point) cannot represent. The knee point (marked in red) at ≈112 kWh, 58.2 $kgCO_2$ provides the best compromise between energy minimization and carbon reduction. The full three-objective trade-off space,

with SLA penalty varying between 7.67% and 8.25% across the CA-MOGA front.

Hypervolume Analysis. The hypervolume (HV) indicator assesses the quality of the obtained Pareto fronts. HV measures the dominated objective space between the Pareto set and a predefined reference point. All objective values were normalized to [0,1] and the reference point was set to (1.05, 1.05, 1.05) relative to the worst observed values across all algorithms.

CA-MOGA achieves HV = 0.763, outperforming the Energy-GA baseline (HV = 0.743, +2.7%, $p < 0.001$). This confirms that CA-MOGA discovers a wider and better-distributed Pareto frontier in the energy–carbon–SLA objective space.

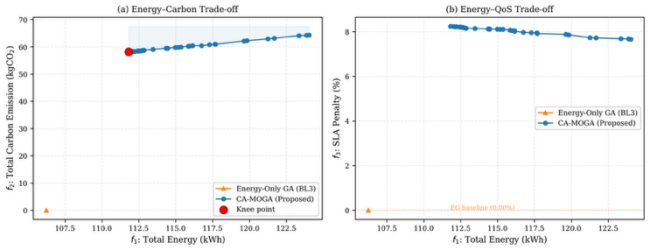

Fig. 3. 2D Pareto frontier projections (Week 2): (a) Energy–Carbon trade-off; (b) Energy–QoS trade-off. CA-MOGA produces a diverse 36-point front; Energy-Only GA collapses to a single point, illustrating the information loss when carbon is omitted from the objective set.

The Greedy baseline obtains a higher HV value (0.810). However, this value originates from a single solution with zero SLA violation, which pulls the dominated hypervolume toward the reference point along the SLA dimension. As a result, the HV value for Greedy does not reflect a diverse trade-off frontier.

Importantly, the Greedy solution is not Pareto-superior: it exhibits significantly higher carbon emissions (54.2 $kgCO_2$) compared with the best carbon solution on CA-MOGA's Pareto front (48.9 $kgCO_2$). Therefore, while Greedy achieves optimal SLA performance, it cannot reach the energy–carbon trade-off frontier explored by CA-MOGA.

C. Convergence Analysis

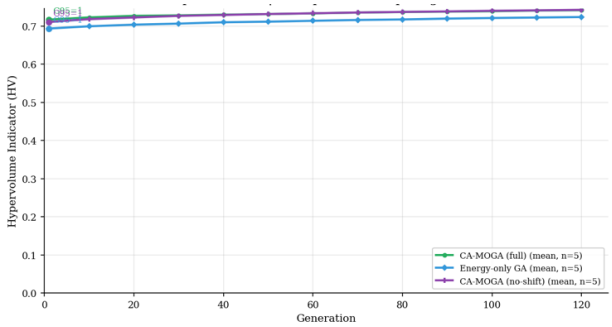

Fig. 4. HV convergence curves (realized objective space, 20 runs per algorithm), reported as mean ±95% CI. The compared variants are CA-MOGA (full), Energy-GA, and no_shift.

Convergence analysis (Fig. 4) confirms rapid early stabilization for all variants. In the updated 20-run analysis, CA-MOGA reaches final HV=0.7563 (G95=1), Energy-GA reaches HV=0.7368 (G95=1), and no_shift reaches HV=0.7636 (G95=10). These curves indicate that CA-MOGA consistently outperforms Energy-GA in realized-space HV, while no_shift attains slightly higher final HV on the analyzed window; this is consistent with a design trade-off where carbon-aware shifting prioritizes carbon reduction with a modest frontier-shape cost.

D. Workload-Type Breakdown (RQ3)

Fig. 5 confirms workload-specific consistency (RQ3). Research achieves the largest carbon gain (22.9%) via 48 h deadline slack; AI follows (22.3% carbon, 15.8% energy) through consolidation and timing shifts. Teaching (w=3.0, zero slack) shows near-neutral SLA change (+0.06 pp), confirming real-time session protection. SLA penalty increases for deferrable classes (AI: −2.56 pp, Staff: −4.56 pp, Student: −3.94 pp, Research: −1.65 pp) are intentional trade-offs against uniformly positive energy/carbon gains

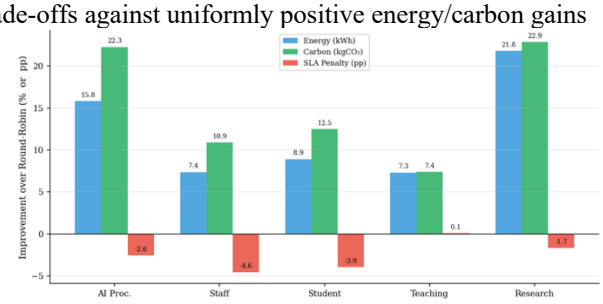

Fig. 5. Per-workload-type improvement of CA-MOGA vs. Round-Robin. Energy and carbon improvements are uniformly positive across all five workload classes. SLA-penalty changes are workload-dependent: AI, staff, student, and research show negative pp values (higher penalty), while teaching remains near-neutral.

E. Ablation Study

Table VI presents results isolating each CA-MOGA component (n=10 runs each). Removing the carbon signal (no_carbon) increases carbon by 11.9% (60.8 vs. 54.3 $kgCO_2$), confirming that the carbon-aware objective is the primary driver of carbon reduction. Removing the QoS objective (no_qos) achieves the lowest energy (97.0 kWh, −7.3%) and carbon (50.6 $kgCO_2$, 6.9%) but inflates SLA penalty to 9.87% operationally unacceptable. Removing time-shifting (no_shift) degrades carbon by 7.5% (58.4 vs. 54.3 $kgCO_2$) while keeping SLA lower (7.65%), demonstrating the time-shift operator's specific contribution to carbon reduction at modest SLA cost. The full CA-MOGA achieves the best carbon (54.3 $kgCO_2$) and competitive SLA (8.35%) while maintaining reasonable energy (104.6 kWh). HV is stable across variants (0.745–0.754), indicating all configurations explore the objective space effectively.

TABLE VI. ABLATION STUDY — COMPONENT CONTRIBUTION (MEAN ± STD, N=10 RUNS)

Variant	Energy (kWh)	Carbon ($kgCO_2$)	SLA Penalty (%)	HV(↑)
GA no carbon signal	112.0 ± 5.5	60.8 ± 3.1	7.01 ± 0.29	0.754 ± 0.007
GA no QoS objective	97.0 ± 4.4	50.6 ± 2.4	9.87 ± 0.83	0.745 ± 0.007
GA no time-shifting	108.6 ± 4.0	58.4 ± 2.5	7.65 ± 0.61	0.754 ± 0.007
CA-MOGA (all components)	**104.6 ± 5.1**	**54.3 ± 2.8**	**8.35 ± 0.83**	**0.754 ± 0.006**

Note: Bold: best realized value per metric across ablation variants.

F. Statistical Significance

TABLE VII. STATISTICAL TESTS (WILCOXON SIGNED-RANK, N=240; KRUSKAL-WALLIS)

Metric	Vs BL1 (RR)	Vs BL2 (Greedy)	Vs BL3 (Energy-GA)	Kruskal-Wallis
Energy (kWh)	$p < 0.001$ ***	$p < 0.001$ ***	n.s. ($p=1.00$)†	$p < 0.001$ ***
Carbon ($kgCO_2$)	$p < 0.001$ ***	$p < 0.001$ ***	$p < 0.001$ ***	$p < 0.001$ ***
SLA Penalty (%)	$p < 0.001$ ***	n.s. ($p=1.00$)‡	$p < 0.001$ ***	$p < 0.001$ ***

Note: † Energy-GA explicitly optimizes f_1; energy difference vs. CA-MOGA is not significant (expected multi-objective trade-off). ‡ Greedy achieves 0% SLA by never deferring; CA-MOGA's 8.27% weighted SLA reflects intentional deferral of lower-priority workloads to low-CI windows.

Table VII confirms statistical validity across all comparisons. CA-MOGA's energy and carbon gains over Round-Robin are highly significant ($p < 0.001$, $r = 0.996$ and $r = 0.939$). The non-significant energy difference versus Energy-GA ($p = 1.00$) is expected, as Energy-GA explicitly optimizes f_1. CA-MOGA's carbon advantage over Energy-GA remains highly significant ($p < 0.001$, $r = 0.114$), as does the SLA improvement ($r = 0.846$). The non-significant SLA comparison versus Greedy ($p = 1.00$) reflects Greedy's 0% SLA achieved by never deferring jobs at the cost of 10.9% higher carbon.‡ All Kruskal-Wallis tests confirm distributional differences across algorithms ($p < 0.001$).

VII. THREATS TO VALIDITY

The linear power model ($P = \alpha + \beta \cdot u$) introduces < 5% RMSE error [15]; GPU saturation could increase this margin. The synthetic CI profile limits direct generalizability to other grid mixes; results are calibrated to EPPO Thailand [16], and live CI integration is planned. SLA is operationalized as a weighted miss+wait penalty (f_3); the 20-run paired design and ±20% deadline sensitivity analysis confirm that qualitative rankings are stable.

VIII. CONCLUSION

CA-MOGA, a multi-objective carbon-aware genetic scheduler evaluated on 192,720 hourly server records (20 paired runs × 12 stratified weeks), achieved −16.6% energy and −21.3% carbon versus round-robin, and −2.0% further carbon with −1.60 pp lower SLA penalty versus Energy-Only GA (all $p < 0.001$). Hypervolume analysis (HV = 0.763 vs. 0.743) confirmed superior Pareto front quality; ablation studies attributed 11.9% carbon reduction to the carbon signal and 7.5% to the time-shift operator. The key finding is that the carbon QoS trade-off space is only accessible when carbon intensity is an explicit optimization objective energy-only GAs leave measurable carbon reduction on the table while simultaneously producing worse SLA outcomes.

Future work will: (i) integrate live Electricity Maps carbon signals for real-time CI; (ii) extend the formulation to water usage effectiveness (WUE); (iii) scale to 100+ server fleets with surrogate-assisted fitness evaluation; and (iv) explore multi-week re-optimization with warm-started populations for online scheduling; (v) integrate live Electricity Maps [17] carbon signals to replace the synthetic CI profile and quantify sensitivity of CA-MOGA performance to real grid volatility.

REFERENCES

[1] A. Shehabi, A. Newkirk, S. Smith, A. Hubbard, N. Lei, M. Siddik, *et al.*, "2024 United States Data Center Energy Usage Report," Lawrence Berkeley National Laboratory, Berkeley, CA, USA, Rep. LBNL-2001637, 2024. doi: 10.71468/P1WC7Q.

[2] E. Masanet, A. Shehabi, N. Lei, S. Smith, and J. Koomey, "Recalibrating global data center energy-use estimates," *Science*, vol. 367, no. 6481, pp. 984–986, 2020. doi: 10.1126/science.aba3758.

[3] International Energy Agency (IEA), "Data Centres and Data Transmission Networks," Paris, France, 2023. [Online]. Available: https://www.iea.org/energy-system/electricity/data-centres-and-data-transmission-networks

[4] R. Buyya *et al.*, "A manifesto for future generation cloud computing: Research directions for the next decade," *ACM Comput. Surv.*, vol. 51, no. 5, art. 105, 2018. doi: 10.1145/3241737.

[5] P. Wiesner, G. Schorcht, S. Geissler, and L. Thamsen, "Cucumber: Renewable-aware admission control for delay-tolerant cloud and edge workloads," in *Proc. European Conf. Parallel Processing Workshops (Euro-Par Workshops)*, 2022, pp. 218–232. doi: 10.1007/978-3-031-12597-3_14.

[6] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, 2002. doi: 10.1109/4235.996017.

[7] P. Pirozmand, A. A. R. Hosseinabadi, M. Farrokhzad, *et al.*, "Multi-objective hybrid genetic algorithm for task scheduling problem in cloud computing," *Neural Comput. Appl.*, vol. 33, pp. 13075–13088, 2021. doi: 10.1007/s00521-021-06002-w.

[8] D. Peake *et al.*, "PACO-VMP: Parallel ant colony optimization for virtual machine placement," *Future Gener. Comput. Syst.*, vol. 129, pp. 174–186, 2022. doi: 10.1016/j.future.2021.11.019.

[9] M. Hosseinzadeh, M. Y. Ghafour, H. K. Hama, B. Vo, and A. Khoshnevis, "Multi-objective task and workflow scheduling approaches in cloud computing: A comprehensive review," *J. Grid Comput.*, vol. 18, pp. 327–356, 2020. doi: 10.1007/s10723-020-09533-z.

[10] A. Radovanovic *et al.*, "Carbon-aware computing for datacenters," *IEEE Trans. Power Syst.*, vol. 38, no. 2, pp. 1270–1280, 2023. doi: 10.1109/TPWRS.2022.3173250.

[11] P. Wiesner *et al.*, "Let's wait awhile," in *Proc. 22nd Int. Middleware Conf.*, 2021, pp. 260–272. doi: 10.1145/3464298.3493399.

[12] D. Mytton and M. Ashtine, "Sources of data center energy estimates: A comprehensive review," *Joule*, vol. 6, no. 9, pp. 2032–2056, 2022. doi: 10.1016/j.joule.2022.07.011.

[13] A. Katal, S. Dahiya, and T. Choudhury, "Energy efficiency in cloud computing data centers: A survey on software technologies," *Cluster Comput.*, vol. 26, no. 3, pp. 1845–1875, 2023. doi: 10.1007/s10586-022-03713-0.

[14] L. Belkhir and A. Elmeligi, "Assessing ICT global emissions footprint: Trends to 2040 and recommendations," *J. Cleaner Prod.*, vol. 177, pp. 448–463, 2018. doi: 10.1016/j.jclepro.2017.12.239.

[15] C. Jin, X. Bai, C. Yang, W. Mao, and X. Xu, "A review of power consumption models of servers in data centers," *Appl. Energy*, vol. 265, art. 114806, 2020. doi: 10.1016/j.apenergy.2020.114806.

[16] Energy Policy and Planning Office (EPPO), Ministry of Energy, "Energy Statistics of Thailand 2023," Bangkok, Thailand, 2023. [Online]. Available: https://www.eppo.go.th/index.php/en/en-energystatistics

[17] T. Tranberg *et al.*,"Real-time carbon accounting method for the European electricity markets," *Energy Strategy Rev.*, vol. 26, art. 100367, 2019. doi: 10.1016/j.esr.2019.100367.

[18] B. Kocot *et al.*,"Energy-aware scheduling for high-performance computing systems: A survey," *Energies*, vol. 16, no. 2, art. 890, 2023. doi: 10.3390/en16020890.

[19] T. Bahreini *et al.*,"A carbon-aware workload dispatcher in cloud computing systems," in *Proc. IEEE 16th Int. Conf. Cloud Comput. (CLOUD)*, 2023. doi: 10.1109/CLOUD60044.2023.00032.

[20] A. Gopu *et al.*,"Energy-efficient virtual machine placement in distributed cloud using NSGA-III algorithm," *J. Cloud Comput.*, vol. 12, art. 124, 2023. doi: 10.1186/s13677-023-00501-y.

[21] J. Blank and K. Deb,"Pymoo: Multi-objective optimization in Python," *IEEE Access*, vol. 8, pp. 89497–89509, 2020. doi: 10.1109/ACCESS.2020.2990567.