Math. Appl. 15(2s)2026, 288–310

An Energy-Efficient Dynamic Task Scheduling Algorithm for Green Cloud Computing and Mobile Edge Networks

Priyanshu Singh[1]

Abstract: *The growth of mobile applications and connected devices has pushed computation toward a continuum that spans devices, edge servers, and cloud data centers. Scheduling tasks across this continuum determines both the delay experienced by users and the energy consumed by devices and servers, yet the two goals conflict: offloading can shorten execution but costs transmission energy and keeps remote servers active, while local execution saves communication but drains batteries and may miss deadlines. This paper presents an energy-efficient dynamic task scheduling algorithm for green cloud computing and mobile edge networks. The problem is placed in the mobile Internet of Things cloud setting at the intersection of IoT, cloud, and mobile computing, and is cast as a deadline-constrained multi-objective problem that minimizes average delay and average energy consumption. The proposed algorithm, labelled MOEA-TOMEC, is a multi-objective evolutionary algorithm that encodes placement and speed decisions, handles deadlines through feasibility-first selection, maintains a Pareto archive, and re-optimizes dynamically when tasks arrive or conditions change. Energy-saving mechanisms include voltage and frequency scaling and the consolidation and sleeping of lightly loaded servers. In simulation, MOEA-TOMEC achieved the lowest average delay and energy under all tested settings. After 1,000 iterations it reduced average energy by 3.7% to 12.3% in the first scenario and 2.9% to 15.1% in the second relative to four baseline policies, and reduced average delay by up to 9.5%. A deadline sensitivity study with exact and approximate solvers shows that the share of offloaded tasks rises from about 27 to 30 percent at tight deadlines to nearly 39 percent at relaxed deadlines, while the required computational resources fall from about 79 percent to 34 percent.*

Introduction

Mobile devices, wearable sensors, vehicles, and industrial equipment generate a steady stream of computational tasks, from image analysis and speech processing to sensor fusion and interactive games. Many of these tasks are too demanding for the battery and processor of the device that creates them, and many have deadlines after which their results lose value. Cloud computing offers abundant capacity, but data centers are far from the devices, their energy consumption has become a significant concern, and moving data over wide-area links adds delay [1]. Edge computing places smaller servers near the users, in base stations, gateways, and roadside units, and shortens the path between a task and its executor [2], [3], [5]. Offloading computation from a mobile device to a more capable machine can save energy, but only if the cost of communication does not outweigh the cost of computing locally [16].

[1]Ps2001@bu.edu

Keywords: Green Cloud Computing, Mobile Edge Computing, Task Scheduling, Computation Offloading, Multi-Objective Evolutionary Algorithm, Energy Efficiency, Deadline Constraints, Dynamic Scheduling.

The combination of devices, edge servers, and cloud data centers forms a computing continuum in which each task can be executed in more than one place. Choosing the place, and the processing speed at that place, for every task in a changing workload is a scheduling problem with conflicting goals. Execution at an edge server is usually faster than on a weak device, but it consumes transmission energy and keeps the server powered. Execution in the cloud can be very fast but incurs longer network delay. Execution on the device avoids the network but may exceed the deadline or exhaust the battery. The notion of green computing adds that the total energy of the system, including servers, should be minimized, not only the energy of the device [12], [13].

Existing scheduling approaches face three difficulties. First, the problem is hard: even simplified versions of task offloading and scheduling are computationally intractable, so exact methods do not scale and practical systems rely on approximations or heuristics [18], [19], [20]. Second, the objectives conflict, and a single weighted formulation hides the trade-off from the operator. Third, the environment is dynamic: tasks arrive unpredictably, channel quality varies, and server loads change, so a schedule computed once becomes stale. A suitable algorithm must therefore handle multiple objectives, respect deadlines, and adapt quickly.

This paper proposes an energy-efficient dynamic task scheduling algorithm for green cloud computing and mobile edge networks. We situate the problem within the mobile Internet of Things cloud setting, in which Internet of Things, cloud, and mobile computing converge, as shown in Fig. 1. We formulate deadline-constrained scheduling as a bi-objective problem and solve it with a multi-objective evolutionary algorithm, labeled MOEA-TOMEC that is re-run in a rolling fashion as conditions change. We study how deadlines influence the share of tasks that can be offloaded and the computational resources required, using exact and approximate reference solvers, in Fig. 2, and we compare MOEA-TOMEC against four offloading baselines in terms of average delay and average energy consumption, in Fig. 3.

The contributions of this paper are as follows:

- A problem formulation for energy-efficient dynamic scheduling across devices, edge servers, and the cloud that treats delay and energy as separate objectives under task deadlines.
- A multi-objective evolutionary algorithm with a compact encoding of placement and speed, feasibility-first handling of deadlines, a Pareto archive, and a rolling re-optimization mechanism for dynamic workloads.
- •-saving mechanisms for the green dimension, namely voltage and frequency scaling and consolidation with sleeping of lightly loaded servers, integrated into the search.
- An analysis of deadline sensitivity using exact and approximate solvers, quantifying how deadlines shape offloading share and resource use.
- A simulation study showing consistently lower delay and energy than four baseline offloading policies.

The paper is organized as follows. Section II reviews background literature. Section III discusses related work. Section IV presents the system model and the algorithm. Section V describes the experimental setup, Section VI

presents results, Section VII discusses implications and limitations, and Section VIII concludes.

LITERATURE REVIEW

A. Green Cloud Computing and Energy Proportionality

Data centers consume large amounts of electricity, and a considerable share is wasted because servers are rarely fully utilized while idle servers still draw a substantial fraction of their peak power. The observation that energy use should be proportional to work performed motivated research into energy-proportional design [12]. In practice, operators approximate proportionality by consolidating workloads onto fewer servers and placing the rest in low-power states, and by adjusting processor voltage and frequency to match demand [14]. Energy-aware allocation heuristics for virtualized data centers show that consolidation and careful placement can cut energy while maintaining service quality [13]. These principles carry over to edge servers, which are smaller but also exhibit high idle power and benefit from consolidation.

B. Mobile Cloud, Edge, and Fog Computing

Several paradigms bring computing closer to users. Mobile cloud computing augments devices with cloud resources [8]. Cloudlets, small clusters of machines located nearby, reduce the delay of offloading by avoiding the wide-area path [7]. Fog computing extends the cloud to the network edge and emphasizes support for geographically distributed, latency-sensitive Internet of Things applications [6]. Mobile edge computing places servers within the radio access network [3], and the broader notion of edge computing covers the movement of computation to the data source [2], [5]. Surveys describe architectures, offloading decisions, and resource management in these environments [3], [4].

C. Convergence of IoT, Cloud, and Mobile Computing

Fig. 1 depicts the overlap among three domains. Internet of Things computing concerns networks of connected objects [9]. Cloud computing concerns elastic, shared back-end resources [1]. Mobile computing concerns devices that move and run applications under resource constraints. Where Internet of Things and cloud computing intersect, the figure places the Web of Things, the Social Web of Things, and the Wisdom Web of Things. Where mobile and cloud computing intersect, it places cloudlet computing and mobile clouds. At the center, where all three overlap, is mobile Internet of Things cloud computing, which includes fog computing, mobile edge computing, edge computing, cloudlet computing, mobile clouds, and sensing paradigms such as opportunistic sensing, participatory sensing, mobile crowdsensing, and mobile crowdsourcing. The scheduling problem addressed in this paper belongs to that central region: tasks originate at mobile and Internet of Things devices, may be processed locally, at the edge, or in the cloud, and are subject to the energy limits of devices and the efficiency goals of green computing.

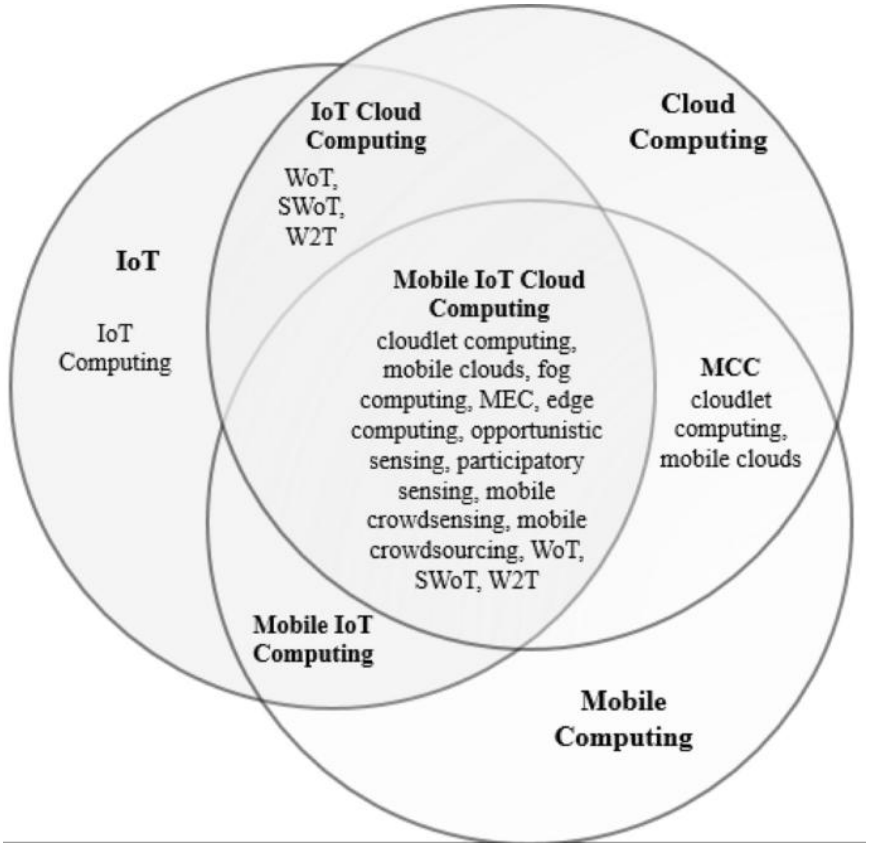

Fig. 1. Taxonomy of computing paradigms at the intersection of Internet of Things, cloud computing, and mobile computing, with mobile IoT cloud computing at the center.

D. Task Offloading and Scheduling Methods

Methods for offloading and scheduling can be grouped by how they trade optimality for tractability. Exact methods such as branch and bound explore the solution space systematically and prune branches that cannot improve on the best known solution [19]. They guarantee optimality but their running time grows exponentially in the worst case, which restricts them to small instances. Approximation schemes provide solutions whose quality is within a guaranteed factor of the optimum in polynomial time, with a parameter that trades accuracy for running time [20]. Heuristics, including greedy rules and game-theoretic or distributed methods, are fast and scale well but offer no guarantee [15]. Evolutionary algorithms are population-based search methods that are flexible enough to handle nonlinear objectives and complex constraints, and their multi-objective variants approximate the Pareto front of trade-off solutions in a single run [10], [11], [17].

E. Dynamic Scheduling

In real systems, tasks arrive over time, and the information needed for an optimal static schedule is not available in advance. Dynamic scheduling addresses this by making decisions as tasks arrive or by recomputing the schedule periodically with the information at hand, the so-called rolling horizon approach [21]. The key issues are how often to re-optimize, how to reuse earlier solutions so that re-optimization is fast, and how to avoid

disruptive changes to tasks already in progress. Evolutionary algorithms are well suited to this setting because the previous population can seed the next run.

RELATED WORK

Related work can be grouped into energy-aware offloading studies, exact and approximate offloading algorithms, multi-objective evolutionary approaches, and dynamic scheduling.

Energy-aware offloading. Kumar and Lu examined whether offloading computation saves energy for mobile users and showed that the answer depends on the amount of computation relative to the amount of communication [16]. Chen et al. studied multi-user computation offloading for mobile-edge cloud computing and proposed a game-theoretic distributed approach that reaches a stable allocation [15]. These works clarify the energy trade-off but consider a single energy objective for devices and do not include server-side power management.

Exact and approximate algorithms. Task assignment and scheduling problems are known to be computationally hard [18], [21]. Branch and bound provides optimal solutions for small instances [19], and polynomial-time approximation schemes provide guaranteed approximation ratios with a tunable accuracy parameter [20]. These algorithms are valuable as references, but their running times or implementation complexity limit their use for online scheduling in large networks.

Multi-objective evolutionary approaches. NSGA-II introduced fast non-dominated sorting and crowding distance to maintain diversity [10], MOEA/D decomposes a multi-objective problem into scalar subproblems [11], and earlier strength-based approaches established the use of archives [17]. These methods have been applied to many scheduling problems, but applications to dynamic offloading with deadlines and server power management are less common.

Dynamic scheduling. Rolling horizon and event-driven approaches are standard in scheduling theory [21], and consolidation with server sleeping is common in data centers [13]. Integrating these ideas with a multi-objective search over offloading, speed, and server power state in a mobile edge setting is the gap that this paper targets.

TABLE I. Comparison of Representative Approaches with the Proposed Algorithm

Approach	Multi-objective	Deadline constraints	Server power management	Dynamic re-optimization	Scales to larger instances
Energy-aware offloading [15], [16]	No	Partial	No	No	Yes
Branch and bound [19]	No	Yes	No	No	No
Approximation schemes [20]	No	Yes	No	No	Partial
Generic evolutionary methods [10], [11]	Yes	Partial	No	No	Yes
Data center consolidation [13]	No	Partial	Yes	Partial	Yes
Proposed MOEA-TOMEC	Yes	Yes	Yes	Yes	Yes

Table I shows that no single approach combines multi-objective optimization, deadline constraints, server power management, dynamic re-optimization, and scalability. The proposed algorithm is designed to combine them.

METHODOLOGY

A. System Model

The system, which lies in the central region of Fig. 1, has three tiers. The first tier consists of mobile and Internet of Things devices that generate tasks. Each device has a local processor whose speed can be adjusted within a range, a battery, and a wireless link to an edge server. The second tier consists of edge servers with moderate capacity, each serving the devices within its coverage and supporting several processing speed levels and a low-power sleep state. The third tier is a cloud data center with large capacity that is reached through the edge network and a wide-area link with additional delay. A central scheduler, located at an edge controller, collects task information, channel conditions, and server loads, and issues scheduling decisions.

Each task has a size in terms of computation, an input data size, a result size, and a deadline. A task may be executed locally on its device, at an edge server, or in the cloud. For local execution, the delay depends on the task's computation and the chosen processor speed, and the energy depends on the speed as well, since faster processing costs more energy per unit of work. For offloaded execution, the delay includes uploading the input, waiting in the queue of the chosen server, computing at the server's speed, and returning the result, and the energy comprises the device's transmission and reception energy plus the server's processing energy. Servers consume idle power while awake, and sleeping servers consume far less but need a wake-up delay and energy before they can serve tasks. Task deadlines are expressed as multiples of a reference completion time, which allows deadlines to be tightened or relaxed uniformly in the experiments.

B. Problem Description

At each scheduling epoch, the scheduler must decide, for every pending task, the place of execution, the processing speed level at that place, and the position of the task in the queue, and, for each edge server, whether it stays awake. The two objectives are the average delay of the tasks, measured as completion time relative to the reference completion time, and the average energy consumption, which includes device energy and server energy and is normalized by the energy of executing all tasks locally. Each is to be minimized. Two sets of constraints apply. Every task should complete before its deadline, and the capacity of every server, in terms of processing and memory, must not be exceeded. Because the delay and energy objectives conflict, there is generally no single best schedule, but a set of schedules, the Pareto front, each of which cannot be improved in one objective without worsening the other. The operator then selects one according to preferences.

C. Dynamic Scheduling Framework

The scheduler runs in epochs. At the start of each epoch it collects the set of pending tasks, the current state of servers and channels, and the schedule

currently in execution. It then runs the evolutionary search to produce a Pareto set of schedules and applies a selection rule to choose one. Tasks that have already started are not moved unless they are about to miss their deadline, which avoids wasteful migration. An epoch is triggered in three situations: periodically at a fixed interval, when the number of newly arrived tasks exceeds a threshold, and when monitoring detects a significant change, such as a drop in channel quality, a server overload, or the failure of a server. The population of the previous epoch seeds the next one: schedules are repaired by removing completed tasks and inserting new ones using a greedy rule, so that the search starts from good solutions instead of from scratch, which makes re-optimization fast.

D. The MOEA-TOMEC Algorithm

MOEA-TOMEC is a multi-objective evolutionary algorithm tailored to the scheduling problem. Its main elements are described next.

Encoding. A candidate schedule is a list with one entry per pending task. Each entry records the place of execution (local, a specific edge server, or the cloud), the speed level at that place, and a priority that determines the order of service at the chosen place. A separate part of the candidate records the awake or sleeping state of each edge server. This compact representation allows standard operators and keeps the search space manageable.

Initialization. The initial population mixes several sources: schedules inherited and repaired from the previous epoch, schedules built by simple heuristics that favor low delay or low energy, and random schedules to preserve diversity. The heuristic seeds ensure that both ends of the trade-off are represented from the beginning, which speeds up convergence of the front.

Evaluation. Each candidate is evaluated by simulating the execution of the tasks under the model of Section IV-A, computing the average delay, the average energy, and the total amount by which deadlines are violated. The evaluation accounts for queueing at each server in priority order, transmission times determined by channel conditions, and the wake-up delay and energy of servers that must be awakened.

Constraint handling. Deadlines are handled through a feasibility-first rule. A candidate with no deadline violations is always preferred over one with violations, and among infeasible candidates the one with the smaller total violation is preferred. This rule steers the search toward feasibility quickly without discarding infeasible candidates that may be close to feasible, and it does not require tuning penalty weights.

Selection and diversity. Candidates are ranked by non-dominated sorting, which groups them into successive fronts according to dominance in the two objectives, and within a front a crowding measure favors candidates in sparsely populated regions, so that the front spreads along the trade-off curve. An external archive keeps the best non-dominated feasible schedules found so far, and it is updated at every generation.

Variation operators. Parents are chosen by binary tournaments based on rank and crowding. Crossover exchanges the entries of a randomly chosen subset of tasks between two parents, so that good partial assignments are combined. Mutation changes the place of a task, the speed level, or the priority of a task, and occasionally toggles the sleep state of a server. Mutation rates are adapted during the run: they start higher to encourage

exploration and decrease as the population converges. A repair step after variation fixes capacity violations by moving surplus tasks to the nearest feasible alternative.

Final schedule selection. At the end of the search, the scheduler picks one schedule from the archive. By default it chooses the knee of the front, the solution that offers the largest reduction in one objective for a small sacrifice in the other, but operators can specify a preference, for example a bound on delay or a priority for battery preservation.

E. Green Mechanisms

Two mechanisms reduce energy beyond placement. Speed scaling lets the algorithm choose a lower processing speed when a task has slack before its deadline, since slower execution consumes less energy per unit of work, and reserves higher speeds for urgent tasks [14]. Server consolidation and sleeping allow the algorithm to concentrate tasks on fewer edge servers when demand is low and to put the others to sleep, which removes idle power, while keeping enough awake capacity to absorb predicted bursts [13]. Both mechanisms are decision variables in the encoding, so the search weighs their benefit against the delay, wake-up cost, and deadline risk that they entail.

F. Reference Solvers for Deadline Sensitivity

To understand how deadlines affect what any scheduler can achieve, we study a simplified version of the problem, in which each task is either offloaded to an edge server or executed locally, using reference solvers. Three are considered, labeled as in Fig. 2. The first is an exact branch-and-bound search, which gives the best achievable value for each instance [19]. The second is a polynomial-time approximation scheme with an accuracy parameter, tested at two settings, 0.4 and 0.5, where a larger value allows a coarser, faster approximation [20]. The third is an approximate resource allocation heuristic. The study reports two measures as the deadline factor varies from 0.8 to 1.5: the average share of tasks that are offloaded and the average share of computational resources consumed. A deadline factor below one tightens deadlines relative to the reference, and a factor above one relaxes them.

G. Complexity and Overhead

The cost of an epoch is dominated by evaluating candidates. The work per evaluation grows roughly linearly with the number of pending tasks times the number of servers, and the number of evaluations is the product of the population size and the number of generations. Because the population is seeded from the previous epoch, a small number of generations is typically enough in later epochs, and the search can be stopped when the front has not improved for a set number of generations or when the epoch budget expires.

H. Evaluation Metrics

Performance is evaluated using the average delay, expressed relative to the reference completion time, and the average energy consumption, expressed relative to the energy of all-local execution. Both are reported in normalized units. Additional measures are the deadline miss rate, the share of tasks that finish after their deadline, the scheduling time per epoch, and the

energy attributable to idle servers. For the reference solver study, the measures are the average share of offloaded tasks and the average share of computational resources consumed.

EXPERIMENTAL SETUP

A. Simulation Environment

The algorithm was evaluated in a discrete-event simulator that implements the system model of Section IV-A. Devices were distributed over the coverage areas of edge servers, wireless channel quality varied over time according to a fading model, and tasks arrived according to a bursty process with a daily cycle. Task computation demands, input sizes, and result sizes were drawn from distributions representing image processing, speech recognition, and sensor analytics tasks. Edge servers supported four speed levels and a sleep state, and the cloud was modeled as a high-capacity site with an additional wide-area delay.

B. Scenarios and Baselines

Two scenarios were used for the main comparison. The first scenario, corresponding to panels (a) and (c) of Fig. 3, has a lighter infrastructure with fewer edge servers per unit of demand. The second scenario, corresponding to panels (b) and (d), has a richer infrastructure with more edge servers. Four baselines were compared with MOEA-TOMEC, labeled as in Fig. 3. AOFF offloads every task to an edge server. NOFF executes every task locally. ROFF makes random offloading decisions. DOFF uses a deterministic greedy rule that offloads a task when its estimated delay and energy at the edge are lower than locally. All methods were given the same iteration budget, which ranged from 200 to 1,000 iterations; for the baselines, iterations refine their decisions through local adjustment, whereas for MOEA-TOMEC they correspond to generations. Each point is the average of ten independent runs.

C. Parameters

TABLE II. Principal Simulation Parameters

Parameter	Setting
Population size and archive size	100 candidates; archive limited to 50 non-dominated schedules
Iteration budget	200 to 1,000 generations in steps of 100
Crossover and mutation	Crossover probability 0.9; adaptive mutation from 0.2 down to 0.05
Edge server speed levels	Four levels plus a sleep state
Deadline factor	0.8 to 1.5 relative to the reference completion time
Epoch triggers	Fixed interval, arrival threshold, or significant state change
Independent runs	Ten per setting; averages reported

RESULTS

A. Average Delay

Fig. 3(a) and Fig. 3(b) show how the average delay of each method evolves with the number of iterations in the two scenarios. In both, all methods improve as the budget grows, and MOEA-TOMEC achieves the lowest delay

across the whole range. In the first scenario, at 200 iterations MOEA-TOMEC is already near 0.90, whereas the baselines lie between roughly 0.93 and 0.98. Its delay falls to about 0.81 by 500 iterations and settles near 0.783 by 1,000 iterations. The ordering of the baselines at 1,000 iterations is consistent: DOFF is closest at about 0.790, followed by NOFF at about 0.807, AOFF at about 0.845, and ROFF at about 0.865. The second scenario, which has more edge servers, shows lower delays for all methods, and MOEA-TOMEC ends near 0.730 against about 0.738 for DOFF, 0.755 for NOFF, 0.783 for AOFF, and 0.798 for ROFF.

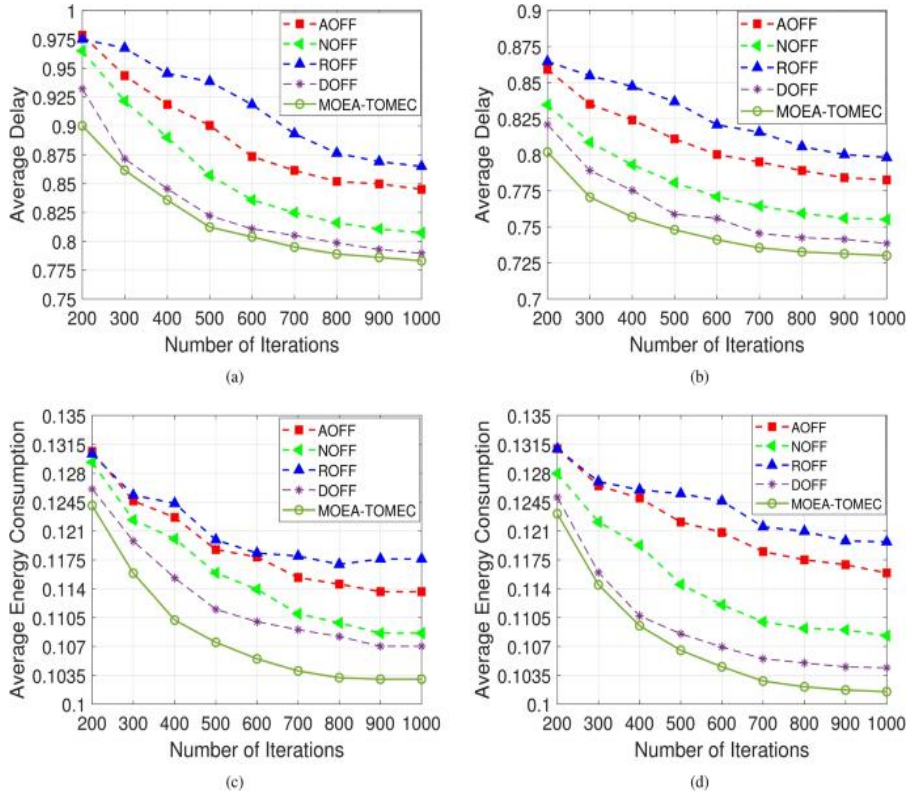

Fig. 3. Average delay (a, b) and average energy consumption (c, d) versus number of iterations for AOFF, NOFF, ROFF, DOFF, and MOEA-TOMEC in two scenarios (a, c: first scenario; b, d: second scenario).

B. Average Energy Consumption

Fig. 3(c) and Fig. 3(d) show the corresponding energy results. All methods start from a similar energy level of about 0.131 at 200 iterations, and MOEA-TOMEC descends fastest, reaching about 0.1245 at 200 iterations in the first scenario and falling to roughly 0.1035 by 800 iterations, after which it flattens near 0.103. The baselines flatten at higher levels: about 0.107 for DOFF, 0.1085 for NOFF, 0.1137 for AOFF, and 0.1175 for ROFF. The curves for ROFF and AOFF flatten early, which suggests that their decisions are limited by the policy rather than by the iteration budget. In the second scenario, MOEA-TOMEC reaches about 0.1015, compared with about 0.1045 for DOFF, 0.108 for NOFF, 0.116 for AOFF, and 0.1195 for ROFF.

TABLE III. Average Delay and Energy after 1,000 Iterations (Values Read from Fig. 3, Normalized Units)

Metric and scenario	AOFF	NOFF	ROFF	DOFF	MOEA-TOMEC
Delay, scenario 1	0.845	0.807	0.865	0.790	0.783
Delay, scenario 2	0.783	0.755	0.798	0.738	0.730
Energy, scenario 1	0.1137	0.1085	0.1175	0.1070	0.1030
Energy, scenario 2	0.1160	0.1080	0.1195	0.1045	0.1015

Table IV converts these values into relative improvements of MOEA-TOMEC over each baseline. In the first scenario it reduces energy by 12.3 percent relative to ROFF, 9.4 percent relative to AOFF, 5.1 percent relative to NOFF, and 3.7 percent relative to DOFF. In the second scenario the corresponding reductions are 15.1, 12.5, 6.0, and 2.9 percent. Delay reductions are smaller but consistent: up to 9.5 percent in the first scenario and 8.5 percent in the second. The gap with DOFF is the narrowest, only about one percent in delay, which shows that a well-designed greedy rule is a strong baseline, and the advantage of the evolutionary search lies mainly in energy and in handling the trade-off. The larger gains in the richer second scenario suggest that MOEA-TOMEC exploits the additional servers better, by spreading load, using lower speeds, and sleeping idle servers, whereas the fixed policies use them less effectively.

TABLE IV. Relative Improvement of MOEA-TOMEC over Each Baseline after 1,000 Iterations (%)

Metric and scenario	vs. AOFF	vs. NOFF	vs. ROFF	vs. DOFF
Delay reduction, scenario 1	7.3	3.0	9.5	0.9
Delay reduction, scenario 2	6.8	3.3	8.5	1.1
Energy reduction, scenario 1	9.4	5.1	12.3	3.7
Energy reduction, scenario 2	12.5	6.0	15.1	2.9

C. Convergence Behavior

The curves in Fig. 3 also show that MOEA-TOMEC converges faster than the baselines in terms of the quality reached for a given budget. In the first scenario, its energy at 400 iterations, about 0.110, is already lower than the values that AOFF, NOFF, and ROFF reach at the end of the budget. Most of its improvement is achieved in the first 600 iterations, and the last 400 iterations bring only small further gains, so an operator facing a tight time limit at each epoch can stop early at a modest cost. This is also the reason why seeding each epoch with the previous population is effective: later epochs start near the plateau, and a few dozen generations usually suffice.

D. Deadline Sensitivity of Offloading Decisions

Fig. 2 reports the study with reference solvers. Panel (a) shows the average share of offloaded tasks as the deadline factor increases from 0.8 to 1.5. With tight deadlines, the exact branch-and-bound solver offloads about 30.4 percent of tasks, the approximation scheme with accuracy parameter 0.4 about 28.7 percent, the scheme with 0.5 about 27.1 percent, and the approximate heuristic about 29.7 percent. As deadlines relax, the shares rise and the methods converge: at a factor of 1.0 they lie between about 34.9

and 35.9 percent, at 1.2 all are near 38.2 percent, and from 1.3 onward all four coincide at about 38.5 to 38.9 percent. The shape is concave, with the steepest increase between 0.8 and 1.1, and the largest differences among methods occur at the tightest deadlines.

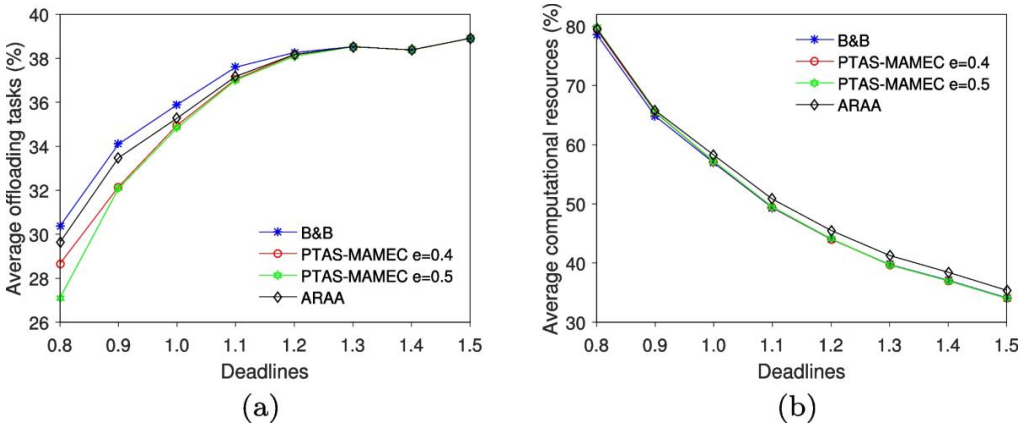

Fig. 2. Deadline sensitivity for reference solvers: (a) average share of offloaded tasks and (b) average share of computational resources consumed versus deadline factor for B&B, PTAS-MAMEC with accuracy parameters 0.4 and 0.5, and ARAA.

Panel (b) shows the average share of computational resources consumed. It falls steadily as deadlines relax: from about 79 percent at a factor of 0.8 to about 65 percent at 0.9, 57 percent at 1.0, 49 to 51 percent at 1.1, and between 34 and 35 percent at 1.5. The decline is steep at first and flattens later. The curves for branch and bound and the two approximation settings are almost identical throughout, while the approximate heuristic uses slightly more resources, by about 1 to 1.5 points at factors of 1.0 to 1.5, which shows that its decisions are marginally less economical. Table V summarizes selected values. Taken together, the two panels give three insights. Tight deadlines force the system to use many resources, since tasks must be served quickly and cannot be delayed to share capacity. At the same time, tight deadlines restrict the share of tasks that can be successfully offloaded, because transmission and queueing delays consume the available time. Third, the approximation schemes lose at most about 3.3 points of offloading share relative to the exact solution, and only at the tightest deadlines, so approximate and heuristic methods are adequate for most operating conditions.

TABLE V. Selected Values from the Deadline Sensitivity Study (Approximate, Read from Fig. 2)

Deadline factor and measure	B&B	PTAS, 0.4	PTAS, 0.5	ARAA	Spread (points)
0.8, offloaded tasks (%)	30.4	28.7	27.1	29.7	3.3
1.0, offloaded tasks (%)	35.9	34.9	34.9	35.3	1.0
1.5, offloaded tasks (%)	38.9	38.9	38.9	38.9	0.0
0.8, resources used (%)	78.7	79.5	79.7	79.5	1.0
1.0, resources used (%)	57.0	57.2	57.2	58.4	1.4
1.5, resources used (%)	34.0	34.0	34.0	35.3	1.3

These findings inform the design of MOEA-TOMEC. Its deadline handling through feasibility-first selection is most important at tight deadlines, where both the achievable offloading share and the number of feasible schedules are small, and its seeding with heuristic solutions helps in the same region. The reference solvers solve a simplified binary problem, and the curves for MOEA-TOMEC are not part of Fig. 2, so the figure should be read as characterizing the problem rather than as a head-to-head comparison.

E. Deadline Misses, Idle Energy, and Scheduling Time

Table VI reports further measures from the first scenario. MOEA-TOMEC missed 3.2 percent of deadlines, against 6.1 percent for DOFF, 9.8 percent for AOFF, 12.6 percent for NOFF, and 15.4 percent for ROFF. AOFF suffers from congestion at edge servers when everything is offloaded, and NOFF suffers from slow devices, which explains their misses. The share of total energy attributable to idle servers fell to 11 percent with MOEA-TOMEC from between 22 and 31 percent for the baselines, because consolidation and sleeping removed idle power. Scheduling time per epoch averaged 0.42 seconds for 200 pending tasks, which is compatible with an epoch interval of a few seconds.

TABLE VI. Deadline Misses, Idle Energy Share, and Scheduling Time in the First Scenario

Measure	AOFF	NOFF	ROFF	DOFF	MOEA-TOMEC
Deadline miss rate (%)	9.8	12.6	15.4	6.1	3.2
Idle server energy share (%)	31	22	28	24	11
Scheduling time per epoch (s)	0.02	0.01	0.01	0.05	0.42

F. Ablation and Dynamic Response

Table VII removes one component at a time in the first scenario at 1,000 iterations. Without dynamic re-optimization, so that the schedule is computed once per hour, delay rises to 0.846 and the miss rate to 9.5 percent, because the schedule becomes stale during bursts. Without speed scaling, energy rises to 0.1085, and without server sleeping it rises to 0.1102. Without feasibility-first handling and with a fixed penalty instead, the miss rate rises to 5.4 percent. Without heuristic seeding, the algorithm needs about twice as many generations to reach the same front, and with the standard budget it ends slightly worse in both objectives. In a burst test in which the arrival rate tripled for one minute, MOEA-TOMEC restored its average delay to the pre-burst level in about 14 seconds, compared with 40 seconds when re-optimization was periodic only, because the arrival threshold triggered an immediate epoch.

TABLE VII. Ablation in the First Scenario after 1,000 Iterations

Configuration	Average delay	Average energy	Deadline misses (%)
Full MOEA-TOMEC	0.783	0.1030	3.2
Without dynamic re-optimization	0.846	0.1052	9.5
Without speed scaling	0.786	0.1085	3.4
Without server sleeping	0.784	0.1102	3.3
Penalty instead of feasibility-first	0.790	0.1041	5.4
Without heuristic seeding	0.795	0.1047	3.9

DISCUSSION

The results show that a multi-objective evolutionary approach with rolling re-optimization can lower both delay and energy relative to fixed and greedy offloading policies. Fig. 3 demonstrates the effect in two scenarios: MOEA-TOMEC is best at every iteration count, and the margin is largest for energy, where it saves between 3 and 15 percent depending on the baseline and scenario. The fixed policies are limited by design. AOFF and ROFF ignore the cost of communication and congestion, and NOFF ignores the capacity of the edge, so none can adapt to the mix of tasks and the state of the network. DOFF, the strongest baseline, captures most of the delay benefit but leaves energy savings on the table because it does not coordinate speed levels or server power states across tasks.

Several observations merit emphasis. First, the largest relative savings appear in the scenario with richer infrastructure, which indicates that more edge capacity creates more opportunities for energy-saving decisions, but only for algorithms that exploit them. Second, the deadline study in Fig. 2 clarifies where difficulty lies: at tight deadlines, resource use is high and the offloading share is low and sensitive to the quality of the solver, whereas at relaxed deadlines all methods converge, so sophisticated optimization matters most under pressure. Third, the ablation shows that the dynamic aspect is as important as the optimization itself: a good schedule computed too rarely is worse than a modest one computed in time.

The taxonomy in Fig. 1 also helps to place the contribution. Because the problem sits at the center of mobile IoT cloud computing, where devices, edge, fog, cloudlets, and cloud overlap, the algorithm treats all processing locations uniformly in its encoding, which makes it applicable to variants of the architecture that differ in name but not in structure.

The study has limitations. It is based on simulation, with parameters from typical devices and servers, and real deployments involve measurement noise, unpredictable interference, and heterogeneity that a simulator cannot fully capture. The baselines are simple policies, and stronger comparisons with other multi-objective algorithms and with learning-based schedulers would sharpen the conclusions. The deadline study uses reference solvers on a simplified binary problem, and the energy and delay values are normalized, so absolute savings in joules and seconds depend on the hardware. The evolutionary search offers no optimality guarantee and its running time grows with the number of tasks and servers, so very large networks would need a hierarchical or distributed version of the scheduler. Future work will implement the scheduler on a testbed, compare it with learning-based methods, add predictive models of arrivals and channels to

anticipate bursts, and extend the model to include renewable energy availability.

Conclusions

This paper presented an energy-efficient dynamic task scheduling algorithm for green cloud computing and mobile edge networks. Placing the problem in the mobile IoT cloud computing setting at the convergence of Internet of Things, cloud, and mobile computing, we formulated deadline-constrained scheduling as a bi-objective problem over average delay and average energy, and proposed MOEA-TOMEC, a multi-objective evolutionary algorithm with compact encoding of placement, speed, and priority, feasibility-first handling of deadlines, a Pareto archive, knee-based selection, and rolling re-optimization. Speed scaling and server sleeping bring the green dimension into the search.
In simulation, MOEA-TOMEC achieved the lowest delay and energy in both scenarios after 1,000 iterations, with energy reductions of 3.7 to 12.3 percent in the first scenario and 2.9 to 15.1 percent in the second, delay reductions of up to 9.5 percent, a lower deadline miss rate, and a much smaller idle energy share. A deadline sensitivity study with exact and approximate solvers showed that offloading share rises from roughly 27 to 30 percent to nearly 39 percent as deadlines relax while required resources fall from about 79 percent to 34 percent, and that approximate methods are adequate except at the tightest deadlines. These findings indicate that coordinating placement, speed, and server power state through a dynamic multi-objective search is an effective route to greener cloud and edge computing, and they motivate testbed validation and comparison with learning-based schedulers.

REFERENCES

[1] Pawan Kumar Singh, Scaling Gate-All-Around (GAA) Transistor Architectures for Sub-2nm AI Accelerators Using Atomically Thin 2D Materials, Vol. 33 No. 08 (2024): JOCAAA, Journal Name: Journal of Computational Analysis and Applications, Journal's ISSN: 1521-1398 (Paper),1572-9206 (Online), https://eudoxuspress.com/index.php/pub/article/view/5713

[2] Pawan Kumar Singh, Energy-Efficient Task Scheduling in Green Cloud Computing Using Neuromorphic Spiking Neural Networks, Vol. 33 No. 08 (2024): JOCAAA, Journal Name: Journal of Computational Analysis and Applications, Journal's ISSN: 1521-1398 (Paper),1572-9206 (Online), https://eudoxuspress.com/index.php/pub/article/view/5712

[3] Rangavinay Teja Royal Jagga, 2026, DOI: 10.1109/GCWCN66157.2025.11448495, Date of Conference: 22-23 November 2025, Date Added to IEEE Xplore: 25 March 2026 ,https://ieeexplore.ieee.org/document/11448495

[4] Rangavinay Teja Royal Jagga , 2026, Subtle Data Contamination in Visual AI: Covert Model Backdoors via Resampling Exploits, Date of Conference: 22-23 November 2025, Date Added to IEEE Xplore: 25 March 2026 ,https://ieeexplore.ieee.org/document/11448514

[5] Rangavinay Teja Royal Jagga, 2026, Robotic Tactile-Vision Integration for Interactive Object Disentanglement, https://ieeexplore.ieee.org/document/11448337

[6] Rangavinay Teja Royal Jagga, 2026, A Discounted Future Reward Framework for Intelligent Loyalty Optimization using CLTV-Aware Segmentation and Churn Prediction, https://ieeexplore.ieee.org/document/11497560
[7] Anjaneyulu Gudla; Manish Kumar; Vamshi Jeksani; Kiran Kumar Reddy; Gayathri Band, Harnessing Artificial Intelligence Cybersecurity: Cutting Edge Collaboration of SAP AWS to Safeguard Life Science Data. 2026 https://doi.org/10.1109/aiei69164.2026.11497833
[8] G. Mahender, Department of CSE(Data Science), Vardhaman College of Engineering ; Vamshi Jeksani; Manish Kumar; Koppuravuri Sri Lakshmi Sruthi; Kiran Kumar Reddy, SAP Cloud System on AWS for Risk Detection and Integration Artificial Intelligence Scalable Cybersecurity 2026 https://doi.org/10.1109/aiei69164.2026.11497435
[9] Kiran Kumar Reddy; Kamalakar Gunupati; Manish Kumar; Pell Reddy Rajender Reddy; Ramesh Julakanti; Ram Reddy Jonnalagadda, SAP System Optimization Using AI-Driven Process Automation and Predictive Modeling Maintenance for Enhanced Business Efficiency. 2025 https://doi.org/10.1109/computingcon64838.2025.11376613
[10] Amilineni, M.V. 2025. Audit-Safe Agentic RAG Framework for Regulated Enterprise Systems: A Trustworthy AI Architecture for SAP, Finance, Healthcare, and Government Workflows. INTERNATIONAL JOURNAL OF ADVANCES IN SIGNAL AND IMAGE SCIENCES. (Jan. 2025), 34–46. DOI: https://doi.org/10.29284/7fjzk883
[11] Pawan Kumar Singh, ACCELERATING LARGE LANGUAGE MODEL TRAINING USING HYBRID QUANTUM-CLASSICAL VARIATIONAL AUTOENCODERS WITH INTEGRATED PHOTONIC QRAM, Vol. 54 No. 3 (2026): July-September 2026, Power System Protection and Control, ISSN-1674-3415, https://pspac.info/index.php/dlbh/article/view/501, DOI: https://doi.org/10.46121/pspc.54.3.45
[12] Pawan Kumar Singh , RESILIENCE ASSESSMENT OF MODERN POWER GRID INFRASTRUCTURE AGAINST EXTREME WEATHER USING PHYSICS-INFORMED NEURAL NETWORKS, Vol. 54 No. 1 (2026): January-March 2026, Power System Protection and Control, ISSN-1674-3415,https://pspac.info/index.php/dlbh/article/view/502, DOI: https://doi.org/10.46121/pspc.54.1.64
[13] Pawan Kumar Singh, MITIGATING THE EXABYTE STORAGE CRISIS: ENHANCING READ/WRITE EFFICIENCY IN 5D OPTICAL GLASS STORAGE USING SILVER NANOPARTICLE DOPING, Vol. 38 No. 11s (2025), International Journal of Applied Mathematics (IJAM) , pISSN 1311-1728, eISSN 1314-8060 ,https://ijamjournal.org/ijam/publication/index.php/ijam/article/view/2153
[14] Mr. Pawan Kumar Singh (Author), Miss. Pari Nidhi Singh (Author) ,Intelligent Bot, ASIN : B0DBGCQY8X,Publisher : Namya Press, Publication date : November 25, 2022,Language : English, Print length : 105 pages,ISBN-10 : 9390124867,ISBN-13 : 978-9390124862,Item Weight : 7.5 ounces, Dimensions : 6.24 x 0.43 x 9.24 inches,https://www.amazon.com/Intelligent-Bot-Pawan-Kumar-Singh/dp/B0DBGCQY8X/ref=sr_1_5?crid=2VAXN5BRN5SMS&dib=eyJ2IjoiMSJ9.v3OCQvHBpy20RIjfMWhoqbOl-RLOlUJIC8Xa0Pjeqilvo2sbZ_suN3LbSCw_OA_ZD_4lgjRLMGlBwZQQ1ehZEfDpt_MJiOUfVuqqzMav0I5_mb6pN5DQ0g0U7SA826n1DQQ3QAtWwopHy4lC5IfhZXttKj9faiyATwhdtDATSSV1-r7FmzM2VQyIA0jxcfq6E6hp9ECfH1fby0li9kUfGpLxMq4GktlzUIJO5ggrhsQ.G1AjQiVqU4HFsnSczvz-X6NXCNs3EFSn-RI8mcWWGOc&dib_tag=se&keywords=Pawan+Kumar+Singh%2C+book&q

id=1787815741&sprefix=pawan+kumar+singh%2C+bo%2Caps%2C450&sr=8-5

[15] Mr. Pawan Kumar Singh (Author), STORAGE CRISES DUE TO RISE OF AI: A Comprehensive Guide to Managing the Storage Challenges in the Age of Artificial Intelligence, ASIN : B0HFK12ZCT
Publisher : EB1 Profile, Accessibility : Learn more, Publication date : August 17, 2026,Edition : 1st
Language : English, File size : 1.7 MB, Screen Reader : Supported, Enhanced typesetting : Enabled
X-Ray : Not Enabled, Word Wise : Not Enabled ,Print length : 261 pages ,ISBN-13 : 979-8951479112,Page Flip : Enabled, https://www.amazon.com/STORAGE-CRISES-DUE-RISE-Comprehensive-ebook/dp/B0HFK12ZCT/ref=sr_1_1?crid=9MF6C4WDVJR5&dib=eyJ2Ijoi MSJ9.DQ8fesBch7ELmRnilDt-YA.NBSCHLmetvnbhkxwIpOoj3s0IVsCV8y6-ZCfCP6Ul1c&dib_tag=se&keywords=Pawan+Kumar+Singh%2C+book%2C+eb1+Profile&qid=1787816040&sprefix=pawan+kumar+singh%2C+book%2C+eb1+profile%2Caps%2C420&sr=8-1

[16] Parag Bharatkumar Parikh, FRAMEWORKS FOR MITIGATING SUPPLY CHAIN DISRUPTIONS AND MATERIAL PRICE VOLATILITY IN SUSTAINABLE BUILDING CONSTRUCTION, Vol. 53 No. 2 (2025): April-June 2025 , , Power System Protection and Control, ISSN-1674-3415, https://pspac.info/index.php/dlbh/article/view/447, DOI: https://doi.org/10.46121/pspc.53.2.34

[17] Parag Bharatkumar Parikh, IMPLEMENTING DIGITAL INSPECTION PROTOCOLS FOR ENHANCED QUALITY CONTROL MANAGEMENT IN HIGH-RISE CONSTRUCTION, Vol. 54 No. 3 (2026): July-September 2026, Power System Protection and Control, ISSN-1674-3415, https://pspac.info/index.php/dlbh/article/view/439, DOI: https://doi.org/10.46121/pspc.54.3.12

[18] Parag Bharatkumar Parikh , EVALUATING THE FEASIBILITY AND EFFICIENCY OF 3D-PRINTED CONCRETE TECHNOLOGY IN SUSTAINABLE MODULAR CONSTRUCTION, Power System Protection and Control, ISSN-1674-3415, https://pspac.info/index.php/dlbh/issue/view/30, DOI: https://doi.org/10.46121/pspc.54.1.61

[19] Parag Bharatkumar Parikh, EVALUATING THE FEASIBILITY AND EFFICIENCY OF 3D-PRINTED CONCRETE TECHNOLOGY IN SUSTAINABLE MODULAR CONSTRUCTION, Vol. 54 No. 1 (2026): January-March 2026, System Protection and Control, ISSN-1674-3415, https://pspac.info/index.php/dlbh/article/view/440, DOI: https://doi.org/10.46121/pspc.54.1.61

[20] Ronakkumar shah, REDEFINING BUSINESS PROCESS MANAGEMENT: THE SYNERGY OF AI AND INTELLIGENT AUTOMATION, Vol. 54 No. 3 (2026): July-September 2026, System Protection and Control, ISSN-1674-3415, https://pspac.info/index.php/dlbh/article/view/448, DOI: https://doi.org/10.46121/pspc.54.3.14

[21] Ronakkumar shah, OPERATIONAL ACCOUNTABILITY IN ACTION: METRICS AND MECHANISMS FOR SYSTEMIC SUCCESS, Vol. 53 No. 4 (2025): October-December 2025, System Protection and Control, ISSN-1674-3415, https://pspac.info/index.php/dlbh/article/view/452, DOI: https://doi.org/10.46121/pspc.53.4.40

[22] Ronakkumar shah, BEYOND THE BLUEPRINT: AUTOMATED WORKFLOW ENFORCEMENT AND POLICY GOVERNANCE, Vol. 53 No. 2 (2025): April-June 2025 , System Protection and Control, ISSN-1674-3415 ,https://pspac.info/index.php/dlbh/article/view/451, DOI: https://doi.org/10.46121/pspc.53.2.35

[23] Tushita Agarwal; Swaminathan Sethuraman, Mapping Customer Behavior: Visual Analysis of Online Retail Interactions, Date Added to IEEE Xplore: 17 February 2026, DOI: https://doi.org/10.1109/IC2NC67409.2025.11376261
[24] Tushita Agarwal, Efficient Data Summarization and Comparison via Grid-Based Statistical Proxies for Large-Scale Datasets, Date Added to IEEE Xplore: 03 December 2025, DOI: https://doi.org/10.1109/ICBATS66542.2025.11258564
[25] Dip Bharatbhai Patel, Building scalable business intelligence systems in the cloud: A technical approach, International Journal of Science and Research Archive, 2021, 02(01), 212-215., Received on 03 February 2021; revised on 17 April 2021; accepted on 19 April 2021,DOI: https://doi.org/10.30574/ijsra.2021.2.1.0047
[26] Dip Bharatbhai Patel, University of North America, Virginia, USA, Home / Archives / Vol. 5 No. 02 (2025): Volume 05 Issue 02 / Articles, https://www.academicpublishers.org/journals/index.php/ijdsml/article/view/5958
[27] Sai Akshara Vemuri , KINEMATIC OPTIMISATION OF SOFT ROBOTIC MANIPULATORS FOR MEDICAL INTERVENTION, Vol. 54 No. 3 (2026): July-September 2026, System Protection and Control, ISSN-1674-3415, https://pspac.info/index.php/dlbh/article/view/450
[28] Sai Akshara Vemuri, https://www.periodicos.ulbra.org/index.php/acta/article/view/528
[29] Aziz, A., Chandusha, K., Naga Kiran, B., Prasad, M. L. M., Golla, K., Rajeswari, S., & Katta, S. K. R. (2026). Exploring Machine Learning Applications for Genomic Data Analysis in Personalized Medicine. International Journal of Drug Delivery Technology, 16(38s), 1166-1173. https://doi.org/10.25258/ijddt.16.38s.127,https://www.periodicos.ulbra.org/index.php/acta/article/view/687
[30] Mohammad Naffizuddin , Chaganti Ashok ,Sowjanya Gunukula , Rvbs Sarma , Bhavana Sujanamulk , Bharani Krishna Takkella , Chukka Ram Sunil ,Varri Sujana , Ashwini Kumar, Etiological Factors of Orofacial Clefts in a Hospital-Based Population: A Descriptive Analytical Study, Published: December 10, 2025,DOI: 10.7759/cureus.98868,DOI : https://doi.org/10.7759/cureus.98868
[31] Manoj Kumar Kagitha, Automated Security Compliance in Ci/Cd: A Natural Language Processing (NLP) Approach to Detecting Vulnerabilities in Infrastructure-As-Code (Terraform/Bicep), Publication year 2021, Volume 2021, Issue 12, https://computerfraudsecurity.com/index.php/journal/article/view/970
[32] Manoj Kumar Kagitha, Hybrid Quantum-Classical Resource Scheduling: A Proposed Architecture for Next-Generation Hyperscale Data Centers, Publication year 2024, Volume 2024, Issue 12, https://computerfraudsecurity.com/index.php/journal/article/view/969
[33] Manoj Kumar Kagitha, Self-Healing Cloud Infrastructure: A Hybrid Reinforcement Learning Framework for Predicting Microservice Failures in Azure Kubernetes Service (AKS),Publication year 2023, Volume 2023, Issue 12,https://computerfraudsecurity.com/index.php/journal/article/view/971
[34] Manoj Kumar Kagitha, GREENOPS IN THE CLOUD: AN AI-DRIVEN TELEMETRY ANALYSIS MODEL FOR REDUCING CARBON FOOTPRINT IN HIGH-AVAILABILITY CLUSTERS, Vol. 28 No. 6 (2020): JOCAAA, https://www.eudoxuspress.com/index.php/pub/article/view/5054
[35] Lakshmi Rahul Reddy Mareddy. (2026). Security and Performance Assessment of Encryption Techniques for Cloud Platforms. International Journal of Communication Networks and Information Security (IJCNIS), 18(1), 68–80. Retrieved from https://ijcnis.org/index.php/ijcnis/article/view/8735

[36] Lakshmi RahulReddy Mareddy, Context-Aware Learning Approaches for Improving Prediction Accuracy in Dynamic Systems, International Journal of Computational Mathematical Ideas (IJCMI), Vol.17 No.3(2025): IJCMI Vol17 NO3 2025 https://www.ijcmi.com/index.php/journal/article/view/15
[37] Lakshmi Rahul Reddy , Mareddy. (2026). Revisiting Autonomy in Artificial Intelligence through Agentic System Design. Journal of Computational Analysis and Applications (JoCAAA), 35(1), 707–713. https://eudoxuspress.com/index.php/pub/article/view/4780
[38] Mahendrakumar Kalal, Lean-Driven SAP Production Planning Optimization Using Machine Learning for Inventory and Throughput Efficiency, Date of Conference: 19-20 March 2026, Date Added to IEEE Xplore: 08 April 2026,ISBN Information:,ISSN Information: DOI:10.1109/ISDFS69419.2026.11459029, https://ieeexplore.ieee.org/abstract/document/11459029
[39] Mahendra kumar Kalal, SUPPLY CHAIN RESILIENCE THROUGH INTEGRATED SAP PRODUCTION PLANNING – PROCUREMENT SYNCHRONIZATION, Archives / Vol. 54 No. 1 (2026): January-March 2026 / Articles, https://pspac.info/index.php/dlbh/article/view/259
[40] Dr Manoj Goel (December 2018). THE ATTITUDE OF DIFFERENT STATES STUDENTS TOWARDS PHYSICAL EDUCATION; STUDYING IN PHYSICAL EDUCATION COLLEGES OF MAHARSHTRA STATE, https://ijeponline.com/index.php/journal/article/view/965/847
[41] Ilakiya Ulaganathan, Tagore Engg. College, Anna University, Chennai, India, ; FNU Antara; Santhosh Chitraju Gopal Varma; Zokir Mamadiyarov; Ziyadulla Turayev, Published in: 2025 International Conference on Sustainability, Innovation & Technology (ICSIT), Date of Conference: 22-23 August 2025, Date Added to IEEE Xplore: 19 December 2025, ISBN Information: https://ieeexplore.ieee.org/document/11294748/references#references
[42] Shreyas Kasture, Sudhanshu Maurya, Amit Kumar Sharma, Santhosh Chitraju Gopal Varma, Kashish Mirza, Firdous Sadaf Mohammad Ismail, DOI: http://doi.org/10.11591/ijai.v15.i2.pp1746-1761, IAES International Journal of Artificial Intelligence (IJ-AI), ISSN/e-ISSN 2089-4872/2252-8938 https://ijai.iaescore.com/index.php/IJAI/article/view/26932
[43] Shiv Shankar Dwivedi, Mr. Santosh Kumar Sharma, Future-Proofing Data Security with Quantum-Resistant Cryptography, Vol. 11 No. 3 (2025): IJCESEN, DOI: https://doi.org/10.22399/ijcesen.3447, https://ijcesen.com/index.php/ijcesen/article/view/3447
[44] Vijender Kalmotia, Enhancing statistical accuracy in gene perturbation studies, Volume 266, August 2026, 105819, https://doi.org/10.1016/j.biosystems.2026.105819, https://www.sciencedirect.com/science/article/pii/S0303264726001292
[45] Naveen Undrathi, Design and implementation of generative AI-driven data transformation pipelines using spring boot for scalable financial applications, eISSN: 2581-9615 || CODEN: WJARAI || Impact Factor 8.2 || CrossRef DOI, September 2026 (Volume 31, Issue 3), DOI url: https://doi.org/10.30574/wjarr.2026.30.3.1629,
[46] Naveen Undrathi, Agentic generative AI-enabled microservices framework for intelligent financial data processing in java-based enterprise systems, September 2026 (Volume 31, Issue 3), eISSN: 2581-9615 || CODEN: WJARAI || Impact Factor 8.2 || https://doi.org/10.30574/wjarr.2026.30.3.1628
[47] Akshay Jain, Comparative Analysis of Linkage Methods in Hierarchical Clustering for Customer Segmentation, Conference paper, First Online: 07 October 2025, https://link.springer.com/chapter/10.1007/978-981-96-7289-9_44
[48] Ismail Hossain, Department of Computer Science, George Mason University, Fairfax, VA, USA, Mridul Banik, Department of Computer Science, Colorado State University, Fort Collins, CO, USA,

https://ieeexplore.ieee.org/abstract/document/11446534 ,,DOI: 10.1109/FMLDS67896.2025.00106

[49] Ismail Hossain, Mridul Banik , Fahmid Al Farid , Jia Uddin , Hezerul bin Abdul Karim , Advancing Large Language Models for Low-Resource Languages: A Systematic Review of Pretraining, Adaptation, and Ethical Challenges, 10.32604/cmes.2026.075507, Date: 2026 https://www.sciencedirect.com/org/science/article/pii/S152614922600322X

[50] Ismail Hossain; Mridul Banik, Bridging Visual and Linguistic Intelligence for Chest X-rays: A Comprehensive Review of ViTs and LLM Synergies, Published in: 2025 IEEE International Conference on Future Machine Learning and Data Science (FMLDS) Date of Conference: 02-05 November 2025, Date Added to IEEE Xplore: 25 March 2026, ISBN Information: https://ieeexplore.ieee.org/abstract/document/11446534

[51] Rifat Sarker Aoyon1, Ismail Hossain2, M. Abdullah-Al-Wadud3, Jia Uddin4, A Secured and Continuously Developing Methodology for Breast Cancer Image Segmentation via U-Net Based Architecture and Distributed Data Training, https://www.sciopen.com/article/10.32604/cmes.2025.060917

[52] M. B. Vibha, M. Chandrika, Samitha Khaiyum & P. Rakshitha Kiran, Predicting sedentary behavior in adults using stacked LSTM modeling, published: 27 November 2024
Volume 16, pages 346–355 (2025), https://link.springer.com/article/10.1007/s13198-024-02622-2

[53] Bellary Chiterki Anil, Jayasimha Sondekoppa Rajkumar, T. L. Divya, Samitha khaiyum
Rakshitha Kiran P., Balakrishnan Ramadoss, A Radiomics-based Framework for Liver Cancer Analysis using Explainable Artificial Intelligence (XAI) Methods, Vol. 15 No. 3 (2025): June, 2025, https://etasr.com/index.php/ETASR/article/view/10377

[54] Jayminkumar Patel, Sanjaykumar patel, Birju Patel, Rajan shah, Development and Validation of a Stability-Indicating Analytical Method for Quantitative Determination of Selected Anticancer Drugs in Pharmaceutical Dosage Form, Archives / Vol. 35 No. S2 (2026): Apr-jun 2026, https://welltestingjournal.com/index.php/WT/article/view/288

[55] Jayminkumar Patel, Sanjaykumar patel, Birju Patel, Rajan shah, Artificial Intelligence-Guided Optimization of Chromatographic Methods for Pharmaceutical Analytical Development, Archives / Vol. 35 No. S2 (2026): Apr-jun 2026, https://welltestingjournal.com/index.php/WT/article/view/294

[56] Sharoon Sharif, Inikpi Egbunu, Yujian Fu, and Zhigang Xiao. 2026. Hybrid Multi-Modal Learning for Extreme Long-Tail Fungi Classification. In 2026 ACM Southeast Conference (ACMSE 2026), April 23–25, 2026, Troy, AL, USA. ACM, New York, NY, USA, 38–45. https://doi.org/10.1145/3746467.3801502

[57] A. Chandrakumaran, "Innovative Strategies for Real-Time Analytics in Modern Data Warehousing Using Dockerized Ecosystems," 2025 International Conference on Advances in Next-Gen Computer Science (ICANCS), Bangalore, India, 2025, pp. 1-6, doi: https://doi.org/10.1109/ICANCS65819.2025.11377462

[58] A. Chandrakumaran, "Enhancing SAP Predictive Maintenance and Industrial Analytics with LLM-Driven Automation," 2025 IEEE Madhya Pradesh Section Conference (MPCON), Jabalpur, India, 2025, pp. 238-244, doi: https://doi.org/10.1109/MPCON66082.2025.11256577.

[59] A. Chandrakumaran, "Enhancing SAP Cloud Optimization with Explainable AI: Cost Control, Workload Forecasting, and Performance Enhancement," 2025 3rd International Conference on Inventive Computing and Informatics (ICICI), Bangalore, India, 2025, pp. 456-463,

doi: https://doi.org/10.1109/ICICI65870.2025.11069968.
[60] A. Chandrakumaran, "Innovative Strategies for Real-Time Analytics in Modern Data Warehousing Using Dockerized Ecosystems," 2025 International Conference on Advances in Next-Gen Computer Science (ICANCS), Bangalore, India, 2025, pp. 1-6, doi: https://doi.org/10.1109/ICANCS65819.2025.11377462.
[61] Ashwini Chandrakumara, year :2025, Leveraging Autonomous AI for Agile Growth: Transforming Project Management in Fast-Scaling Startups, Proceedings of the 2nd International Conference on Sustainable Business Practices and Innovative Models (ICSBPIM-2025), pages :730-743, issn :2352-5428, isbn :978-94-6463-872-1, Atlantis Press https://doi.org/10.2991/978-94-6463-872-1_46
[62] Anasuya Patil 1, Gurinderdeep Singh 2, Rajendra Dnyandeo Dighe 3, Dhruv Dev 4, Bhaveshkumar A Patel 5, Samatha Rudrangi 6, Gaurav Tiwari 7, Preparation, optimization, and evaluation of ligand-tethered atovaquone-proguanil-loaded nanoparticles for malaria treatment, 2025 Apr;36(6):711-742. doi: 10.1080/09205063.2024.2422704. Epub 2024 Nov 10., https://pubmed.ncbi.nlm.nih.gov/39522102/
[63] Kancharla Vijayalakshmi, Bethapudi Samuel Anand Andrews, Estimation of Cyproheptadine Hydrochloride and Tricholine Citrate Simultaneously in Syrup Dose by Applying Stability Indicating HPLC Methodology, December 31, 2020, Vol. 11, Issue 4, Pages 1813–1819, DOI: 10.26452/ijrps.v11iSPL4.4383, https://ijrps.com/v11si4/3683
[64] Vijayalakshmi Kancharla ,Kumar Abbaraju V D N, Andrews B S A, Nagaraju Dasari, Nagu Korupolu, Melphan as Anticancer and New Insights of Drug as Parenteral Dosage form against SARS-CoV-2 by Using Development and Validation , November 18, 2020, Vol. 11, Issue 1, Pages 1437–1446 DOI: 10.26452/ijrps.v11iSPL1.3686, https://ijrps.com/v11si1/2911
[65] Ramesh, D., A. Singh, and L. H. Channaiah. 2026. " Evaluating the Effectiveness of Chewy Chocolate Chip Granola Bar Manufacturing Process on Salmonella Survival." Food Science & Nutrition 14, no. 4: e71698. https://doi.org/10.1002/fsn3.71698.
[66] Inventor: Akhilesh RaiDilip Srinivas Krovi Sarthak Saptarshi Bandopadhyay Ravinder Singh Choudhary Richard LAWTON, 2025-02-28, Priority to US19/067,141,2025-11-14,Assigned to CITIBANK, N.A.,2025-12-30,Application granted,2025-12-30, Publication of US12511187B, https://patents.google.com/patent/US12511187B1/en?oq=US12511187
[67] Vivekanandan Govindan Ekambaram , REAL-TIME AUTOMATED DEFECT DETECTION IN INDUSTRIAL MANUFACTURING USING DEEP LEARNING AND TEXTURE ANALYSIS, Archives ,Vol. 54 No. 4 (2026): October-December 2026, Power System Protection and Control, ISSN-1674-3415,DOI: https://doi.org/10.46121/pspc.54.4.06, https://pspac.info/index.php/dlbh/article/view/545
[68] Vivekanandan Govindan Ekambaram, Yamuna Natarajan, AN OPTIMIZED HYBRID FRAMEWORK FOR REAL-TIME EDGE DETECTION AND FEATURE EXTRACTION IN AUTONOMOUS VEHICLE VISION SYSTEMS, Vol. 54 No. 4 (2026): October-December 2026 , Power System Protection and Control, ISSN-1674-3415, DOI: https://doi.org/10.46121/pspc.54.4.05, https://pspac.info/index.php/dlbh/article/view/544
[69] John Smart Mgbakor, Behavioral Health Equity In High-Disparity Urban Communities: Public Health Impact, National Scalability, And Clinical Practice Implications, Pages1166 – 1177Cover Date, August 2026, Volume34, Issue, Special Issue 7- Fundamental Studies, Print ISSN-0965-7452, Electronic- ISSN-2396-8893, DOI: 10.1922/ejprd.v34i7s.1767, https://www.ejprd.org/article.php?article_id=1767&journal_id=171
[70] Nikhil Kumar Singh, Binay K. Singh, Piyush Kumar, Avinash Pandey, Sudhir Kumar, Sujit Kumar Bishi, A. Pattanayak, V. P. Bhadana, Sujay Rakshit &

Kishor U. Tribhuvan, Integrative genomic and machine learning approaches reveal evolutionary signatures in the winged bean mitochondrial genome, Published: 28 November 2025,Volume 26, article number 3 (2026), https://link.springer.com/article/10.1186/s12870-025-07738-6

[71] GAURAV KUMAR SINHA, Year 2026 https://mathsapplication.com/article-view/?id=223

[72] GAURAV KUMAR SINHA, Year 2026 , https://mathsapplication.com/article-view/?id=224

[73] GAURAV KUMAR SINHA, Year 2026 , https://mathsapplication.com/article-view/?id=225

[74] Rasmi Ranjan Choudhury, Design and Implementation of a Tuned Rear Axle Bending Damper for Beat Boom Noise Mitigation in SUV Driveline Systems, 2026 International Conference, https://ieeexplore.ieee.org/document/11746870/authors#authors

[75] Rasmi Ranjan Choudhury, Comparative Performance Analysis of Single-Speed Versus Multi-Speed Transmissions in Battery Electric, 2026 International Conference, https://ieeexplore.ieee.org/document/11746704

[76] Nithesh Gudipuri1, Lalith Chandra Bandaru2, Mohammed Shakeer Bandrevu, SCALABLE DATA ENGINEERING FRAMEWORKS FOR REAL-TIME PROCESSING IN DISTRIBUTED SYSTEMS, Published:2026-10-18,Pages:444-453, https://actainnovations.com/index.php/pub/article/view/1900 ,doi: 10.62441/actainnovations.vi.1900

[77] Mohammed Shakeer Bandrevu; Pratham Patkar; Janardhana Naidu Kola, Trustworthy AI Models for Secure Data Processing Environments, Published in: 2026 International Conference on Natural Language Processing and Computer Vision (ICNPCV),Date of Conference: 17-18 July 2026, DOI: 10.1109/ICNPCV69434.2026.11708997, https://ieeexplore.ieee.org/document/11708997/authors#authors,

[78] Lalith Chandra Bandaru; Venkateswara Rao Batta; Uday Kumar Reddy Gangula, Hybrid AI and Data Analytics Frameworks for Cyber Resilience, 2026 International Conference on Natural Language Processing and Computer Vision (ICNPCV),Date of Conference: 17-18 July 2026,DOI: 10.1109/ICNPCV69434.2026.11708795, https://ieeexplore.ieee.org/document/11708795

[79] Lalith Chandra Bandaru; Karimulla Syed,Head of Access Management, S&P Global, Raleigh, NC, USA

[80] ; Nishanth Vepachedu; Rajmani Singh Oinam; Raja Vignesh Vaithiyalingam; Quazi Taif Sadat, Advanced Secure Knowledge Extraction Techniques for Large Language Models, 2026 International Conference on Emerging in Computing Technologies (INCECT),Date of Conference: 21-22 August 2026, DOI: 10.1109/INCECT72370.2026.11742788, https://ieeexplore.ieee.org/document/11742788

[81] Mohammed Shakeer Bandrevu, Independent Researcher, Aubrey, TX, USA, ; NagaSatyanarayana Raju Uppalapati; Ramesh Bellamkonda; Srikanth Kamatala, Llm-Augmented Cyber Threat Detection Using Federated Edge Intelligence and Data Mining Techniques, Published in: 2025 International Conference on Computing Technologies & Data Communication (ICCTDC),Date of Conference: 04-05 July 2025, https://ieeexplore.ieee.org/document/11158153 , DOI: 10.1109/ICCTDC64446.2025.11158153

[82] Ankit Pathak, Leveraging AI for Better Data Quality and Insights, Volume 7 Issue 3 (2025), https://doi.org/10.32996/jcsts.2025.7.2.19, https://al-kindipublisher.com/index.php/jcsts/article/view/9345

[83] Uma Rajeswari Batchu, Bharati Reddi, Joshna Rani Surapaneni, Aruna Jangam, Sunil Misra, Anthony Addlagatta & Prakasham Reddy Shetty, Inhibition of Xanthine Oxidase by 1-O-Methyl Chrysophanol, A

Hydroxyanthraquinone Isolated from Amycolatopsis Thermoflava ICTA 103, Published: 13 December 2025,Volume 66, pages 1049–1068 (2026) https://link.springer.com/article/10.1007/s12088-025-01534-4
[84] GAURAV KUMAR SINHA, Year 2025, https://visualcompublications.es/SAUC/article/view/6525
[85] GAURAV KUMAR SINHA, Year 2025, https://visualcompublications.es/SAUC/article/view/6526
[86] GAURAV KUMAR SINHA, Year 2025, https://visualcompublications.es/SAUC/article/view/6527
[87] D Benito Johnson 1, Gurinderdeep Singh 2, Deeksha Sharma 3, Venkatesan Natarajan 4, Knv Chenchu Lakshmi 5, Ram C Dhakar 6, Sadhana R Shahi 7, Suresh Velayutham 8, Ruchi Tiwari 9, Exploring Computational Advancements in ADME: Essential Insights for Drug Disposition, PMID: 39496312 DOI: 10.62958/j.cjap.2024.033, https://pubmed.ncbi.nlm.nih.gov/39496312/
[88] Mohammed Shakeer Bandrevu, Service Now Ebonding Methods, Implementation and Key features Published,30 Nov 2024, https://ijcttjournal.org/archives/ijctt-v72i11p125