# Efficient Task Allocation in Internet of Things Using Lévy Flight-Driven Walrus Optimization

Yaozhi CHEN
Sichuan Vocational and Technical College, Suining 629000, China

***Abstract*—The rapid growth of the Internet of Things (IoT) has presented a significant challenge in efficiently managing energy-aware task distribution over heterogeneous devices. Optimizing the efficient use of resources in terms of energy consumption is critical when considering IoT device resource-constrained environments. This study proposes a new IoT task distribution resolution mechanism using an Enhanced Walrus Optimization Algorithm (EWOA). EWOA incorporates sophisticated techniques, such as Lévy flight processes and augmented exploration-exploitation, and thus is best suited to complex and dynamic IoT environments. This study proposes an EWOA to assign effective tasks considering device capability compatibility and reduced energy consumption. Simulations over benchmark IoT scenarios validate that the EWOA outperforms current approaches in terms of efficiency in terms of energy consumption, convergence, and robustness. In conclusion, improvements in minimizing energy consumption, enhancing task execution performance, and efficient use of resources in IoT networks have been emphasized significantly. In this work, the EWOA was proven to be an effective tool for IoT NP-hard optimization problem resolution and opens doors for future work in utilizing sophisticated metaheuristic algorithms for use in energy-constrained environments.**

***Keywords*—*Internet of things; energy efficiency; task scheduling; walrus; optimization***

## I. Introduction

The Internet of Things (IoT) is a new technology model providing unimpeded connectivity between various smart gadgets [1]. The interconnected IoT environment comprises sensors, controllers, cameras, alarm panels, smart street lights, IP television, public addresses, and Programmable Logic Controllers (PLC) [2]. IoT enables real-time observation, management, and automation, enhancing security, efficiency, and living standards [3]. There are plenty of real-world implementations of IoT platforms, including smart healthcare [4], intelligent streetlights [5], video observation networks [6], and Supervisory Control And Data Acquisition (SCADA) networks [7].

Moreover, IoT forms the technological backbone of emerging domains such as FinTech, where the integration of real-time connectivity and blockchain technologies is accelerating financial innovation and global adoption [8]. Likewise, security remains a critical concern in IoT-enabled mobile and social networks, where intrusion detection systems (IDS) have been proposed to monitor communication patterns and ensure the integrity of distributed ad hoc environments [9]. As the volume of raw data generated by IoT devices grows, efficient transmission and processing through edge or cloud platforms become essential for real-time responsiveness and sustainable system performance [10].

IoT networks feature a variety of interconnected heterogeneous capabilities in distributed environments [11]. Nevertheless, one of the key challenges in such networks lies in resource constraints, specifically energy shortages. Most IoT entities, including wireless sensor nodes, are powered by batteries and have considerable energy limitations [12]. Minimizing energy consumption during data transmission, computation, and executing activities is imperative for extending network operational life. The importance of intelligent, resource-efficient processing is further emphasized in industrial domains such as smart manufacturing, where deep learning-based systems, like hybrid ensembles of vision transformers and convolutional neural networks have been employed to reduce waste and improve efficiency in steel surface defect detection [13, 14].

One effective mechanism for countering such energy constraints is to allow collaboration in scheduling, where network entities make efficient use of computational capabilities [15, 16]. Nevertheless, IoT scheduling is a challenging problem given the device homogeneity, workloads, and the efficient use of communication channels [17]. With their dynamic nature, IoT environments make such a problem even more challenging; therefore, developing smart scheduling techniques for efficient use of resources and minimizing energy consumption is imperative for an efficient IoT environment [18, 19].

This need is echoed in environmental monitoring applications, where real-time systems must integrate heterogeneous spatiotemporal data for emissions tracking, such as methane leakage in the energy sector. Recent work highlights how data alignment, fusion, and resolution integrity challenges can be addressed through advanced analytics and visualization techniques, reinforcing the role of intelligent scheduling and data-driven decision-making in complex IoT systems [20]. This paper proposes an Enhanced Walrus Optimization Algorithm (EWOA) to resolve the IoT scenario energy efficient scheduling issue. The key contribution of this work is:

- We develop an optimization model that leverages EWO to enhance task scheduling and conserve energy in IoT environments with limited resources.
- We integrate a Lévy flight and adaptive search mechanisms with EWO to enhance its convergence velocity and avert early stagnation in local optima.
- A comprehensive performance analysis is conducted through simulations in MATLAB, comparing EWO with

WOA, PSO, and GA inefficiency in terms of energy, processing time, and network steadiness.

The remainder of this paper is organized as follows. Section II summarizes relevant research on IoT task scheduling. Section III presents the problem statement, IoT task scheduling model, constraints, and objective function. Section IV details the proposed algorithm. Section V outlines and discusses simulation results. Section VI comprehensively discusses the findings, their implications, and comparisons with existing work. Section VII concludes the paper with key observations and presents future research directions.

## II. RELATED WORK

Weikert, et al. [21] proposed a multi-objective scheduling algorithm for IoT network failures, including communication loss, battery depletion, and device failure. An archive-selection mechanism was included for search-space diversity, offering reliable alternative mappings in the case of failure. Performance evaluation via a network simulation model revealed that MOTA maximizes network life, reduces latency, and maximizes availability.

Ren, et al. [22] combined Simulated Annealing (SA) and Particle Swarm Optimization (PSO) algorithms for IoT environments' NP-hard problem of scheduling tasks. PSO's problem of getting trapped in a local optimum is avoided in the proposed scheme by leveraging the SA's feature of exploration. Simulations in MATLAB have proven that the proposed scheme is better than traditional PSO and SA-based approaches in terms of increased efficiency in task execution and optimized consumption of resources in IoT networks.

Bali, et al. [23] incorporated an Artificial Bee Colony (ABC) with the Whale Optimization Algorithm (WOA) to counter ABC's early convergence problem with the Employee Bee and Onlooker Bee phases. MATLAB simulations exhibited significant performance improvements, with 50%, 25%, and 60% improvements in energy efficiency, problem execution time, and overall expense, respectively, over the standalone ABC and WOA algorithms.

Nematollahi, et al. [24] designed a fog computing-based scheduling scheme with a combination of Moth-Flame Optimizer (MFO) and Opposition-Based Learning (OBL) for efficient job scheduling. A blockchain fuels a supporting layer to provide accurate information and prevent system overload imbalances. Python simulations confirmed that the proposed model, OBLMFO, reduced latency by 12.18% and saved 6.22% energy consumption, confirming its efficiency in IoT networks with limited resources.

Nematollahi, et al. [25] proposed an Improved Multi-Objective Aquila Optimization (IMOAO) algorithm for offloading jobs to maximize system response and efficiency. Their algorithm utilizes OBL for diversity in the search space and Pareto front selection for improvement. Task-to-fog-node ratio comparisons showed that IMOAO performed better in terms of failure and response times than PSO, Firefly Algorithm (FA), and Multi-Objective Bacterial Forging Optimization (MO-BFO).

Satouf, et al. [26] developed semi-dynamic real-time scheduling for cloud-fog environments with an adaptive Grey Wolf Optimizer (GWO) for effective IoT job scheduling concerning job duration, availability, and network state. It outperforms traditional scheduling algorithms such as Genetic Algorithm (GA), PSO, and ABC regarding processing time, makespan, and saving energy.

Umer, et al. [27] designed a Multi-Objective Task-Aware Scheduling and Offloading Framework (MT-OSF) for IoT-smart transportation systems. Their mechanism identifies a delayed and computation-intensive mechanism with a high-priority offloader and offloads them with a multi-criterion decision mechanism (AHP-based ranking of fog nodes). Their Task-aware scheduler assigns resources considering node energy, bandwidth, RAM, MIPS power, and the short distance to connected cars.

While existing studies cover several aspects of IoT task distribution, some issues have not yet been addressed, as listed in Table I. Most studies emphasise minimising latency or optimizing resources, and energy efficiency occur afterwards. Metaheuristic algorithms, such as PSO, ABC, and SA, have been proposed with promise, but most suffer from premature convergence and become stuck in local optima.

TABLE I. COMPARISON OF EXISTING APPROACHES

| Study | Optimization techniques | Key advantages | Limitations |
|---|---|---|---|
| [21] | Multi-target task assignment procedure with archiving methodology | Enhances network lifetime and reduces latency | Does not explicitly optimize energy efficiency |
| [22] | Simulated annealing and particle swarm optimization | Overcomes local optima trapping and improves resource utilization | Lacks real-world validation, only MATLAB-based |
| [23] | Whale optimization algorithm and artificial bee colony | Reduces energy consumption by 50% and task execution time by 25% | High computational overhead |
| [24] | Moth-flame optimization and opposition-based learning | Improves task execution efficiency | Limited scalability analysis |
| [25] | Improved multi-objective aquila optimization with pareto front selection | Achieves lower response time and failure rate | Not evaluated in large-scale IoT networks |
| [26] | Cloud-fog task scheduling with adaptive grey wolf optimizer | Optimizes execution time, cost, and energy | Does not address node failures |
| [27] | Multi-objective task-aware offloading with AHP-based multi-criteria scheduler | Reduces response time by 7% and energy by 16% | More suited for transportation use cases |

Additionally, fault-tolerant scheduling algorithms have not yet been researched in detail, with most studies combining real-time reallocation of jobs in the case of failure in one node. Security and robustness can be boosted with multi-objective optimization and blockchain techniques but at a high computational expense and with less analysis for scalability. In this study, EWOA is proposed to bridge these gaps with Lévy flight for search improvement, a multi-objective fitness function for scheduling with consideration for energy, and an adaptable mechanism for fault tolerance in execution.

## III. Problem Formulation and Network Model

The task allocation problem in IoT networks can be effectively represented using a Directed Acyclic Graph (DAG), which models dependencies among tasks and the computational costs associated with execution and communication. A DAG is defined as a graph in its entirety $G = (V, E)$, where $V$ stands for the nodes (tasks), and $E$ denotes the edges (communication dependencies between tasks). Each node corresponds to an individual task, while the edges define the precedence constraints that must be satisfied for execution. Task scheduling in IoT is challenging due to the interdependencies among tasks, requiring efficient execution order management to minimize delays and energy consumption.

In a DAG-based task allocation model, the $j^{\text{th}}$ task cannot commence execution until all its parent tasks (denoted as node $i$) are completed. When a parent task is executed, its child tasks become eligible for execution. A node weight represents the computational cost of executing a task, whereas the communication cost between tasks is represented by an edge weight. Fig. 1 illustrates a streamlined DAG for allocating tasks in an IoT environment, where each task is interconnected based on its execution dependencies. The weights assigned to the edges and nodes capture relationship sequences and the associated computational and communication burdens.

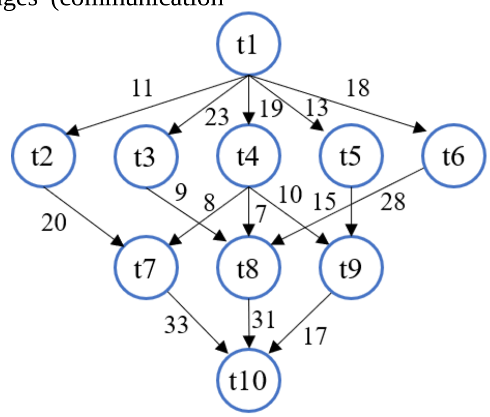


Fig. 1. A streamlined DAG for IoT task allocation.

The IoT architecture used in this study comprises four fundamental layers, each playing a distinct role in data acquisition, transmission, processing, and execution. These layers include sensing, networking, service provisioning, and application, as depicted in Fig. 2. The sensing layer collects data through various smart devices, sensors, and RFID tags. It captures real-time environmental information, which is then processed for decision-making. The networking layer ensures connectivity between IoT devices by utilizing various networking technologies. It facilitates data transmission between perception layer devices and higher processing layers while managing high data traffic efficiently. The service provisioning layer handles data analysis, security, and resource management. It ensures the integrity of collected data, optimizes processing through task allocation, and secures communication pathways. The application layer acts as a user interface, where IoT applications interact with the system. It defines specific real-time requirements, business models, and functional needs of the IoT ecosystem.

The task scheduling mechanism is primarily handled at the service management layer, responsible for assigning computational tasks to appropriate resources while ensuring energy efficiency and minimal latency. The multi-layered architecture of IoT introduces additional complexities, making it necessary to develop efficient task allocation algorithms to optimize resource utilization.

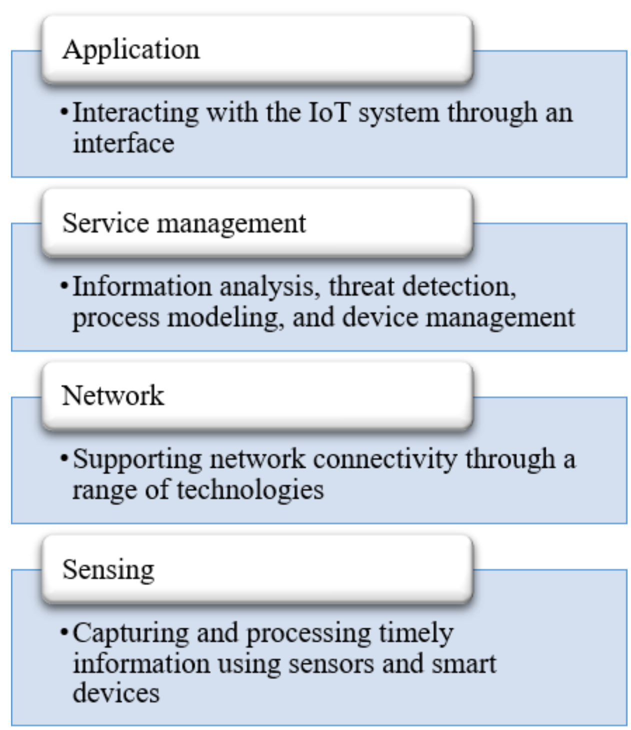


Fig. 2. Adopted IoT architecture.

## IV. Enhanced Walrus Optimization Algorithm

WOA is a metaheuristic optimization technique inspired by the social behaviors of walruses. WOA is designed based on the activities and interactions of walrus populations, including migration, foraging, social bonding, and self-defense mechanisms [28]. These behaviors are mathematically modeled using an optimization framework that balances exploration (global search) and exploitation (local search) in the search space. WOA follows a structured approach comprising four key phases: initialization, danger and safety signals, migration, and reproduction.

The algorithm begins by generating an initial population of candidate solutions, randomly positioned within the defined search space between the upper and lower boundaries of the problem variables. Each solution represents a walrus agent, and their positions are iteratively updated as the algorithm progresses.

WOA utilizes danger and safety signals to model walrus behavior, particularly their response to external threats. The danger signal is computed using Eq. (1).

$$\begin{gathered} Danger\_signal = A \times R \\ A = 2a \\ a = 1 - \frac{t}{T} \\ R = 2 \times r_1 - 1 \end{gathered} \tag{1}$$

The safety signal is defined using Eq. 2.

$$Safety\_signal = r_2 \tag{2}$$

Where $A$ and $R$ determine the intensity of the danger response, with $\alpha$ decreasing linearly from 1 to 0 over time. $r_1$ and $r_2$ are random values between 0 and 1, while $t$ represents the current iteration number and $T$ is the maximum iteration count.

The migration phase is responsible for exploration, allowing walrus agents to move across the search space for better solutions. The position update equation for a walrus agent is given by Eq. (3).

$$X_{i,j}^{t+1} = X_{i,j}^{t} + Migration\_step \tag{3}$$

Where the migration step is calculated using Eq. 4.

$$\begin{gathered} Migration\_step = (X_m^t - X_n^t) \times \beta \times r_3^2 \\ \beta = 1 - \frac{1}{1 + e^{-10(t-0.5T)/T}} \end{gathered} \tag{4}$$

Where $X_m^t$ and $X_n^t$ are randomly selected walrus positions, while $\beta$ is a migration step control factor that adjusts the step size dynamically. The random parameter $r_3$ enhances randomness, allowing better exploration.

During the reproduction phase, walruses adjust their positions based on gender-based social interactions. For female walrus agents, their new position is influenced by both the male walrus and the best-performing solution in the population, as follows:

$$\begin{aligned} female_{i,j}^{t+1} = female_{i,j}^{t} + \alpha \times \left( male_{i,j}^{t} - female_{i,j}^{t} \right) \\ + (1 - \alpha) \times \left( X_{best}^{t} - female_{i,j}^{t} \right) \end{aligned} \tag{5}$$

For juvenile walrus agents, their positions are adjusted using a Lévy flight-based movement strategy as follows:

$$\begin{gathered} juvenile_{i,j}^{t+1} = \left( O - juvenile_{i,j}^{t} \right) \times P \\ O = X_{best}^{t} + juvenile_{i,j}^{t} \times LF \end{gathered} \tag{6}$$

Where $O$ is the safety reference location, $P$ is the danger coefficient associated with juvenile walrus agents, and $LF$ is a Lévy flight-based movement step.

To enhance WOA's efficiency, the EWOA incorporates Lévy flight-based exploration, improving search efficiency and solution diversity. Lévy flights introduce random jumps with heavy-tailed distributions, allowing walrus agents to escape local optima and search over a broader space. The Lévy function used in EWOA is defined using Eq. (7).

$$\begin{gathered} LF = 0.01 \times \frac{u \times \sigma}{|v|^{\frac{1}{\gamma}}} \\ \sigma = \left( \frac{\Gamma(1+\gamma) \times \sin\left(\frac{\pi\gamma}{2}\right)}{\Gamma\left(\frac{1+\gamma}{2}\right) \times \gamma \times 2^{\frac{\gamma-1}{2}}} \right)^{\frac{1}{\gamma}} \end{gathered} \tag{7}$$

Where $u$ and $v$ are random values between 0 and 1, $\gamma$ is the Lévy distribution parameter, and $\Gamma$ denotes the Gamma function. By integrating Lévy flights, EWOA achieves faster convergence, better solution diversity, and more efficient search behavior than the original WOA. Fig. 3 illustrates the flowchart of EWOA.

EWOA is tailored for optimized energy utilization in task scheduling for IoT by leveraging its adaptive exploration-exploitation balance to efficiently allocate computational tasks across heterogeneous IoT devices. EWOA ensures energy-efficient scheduling by dynamically assigning tasks to IoT devices based on computational capacity, energy constraints, and communication overhead. The Lévy flight-based exploration enables the algorithm to search for optimal task-to-node mappings, preventing premature convergence while minimizing energy consumption and execution time.

EWOA integrates a multi-objective optimization approach, where the fitness function considers energy consumption, task execution latency, and workload balancing. The danger and safety signaling mechanism helps prioritize critical tasks, ensuring that time-sensitive operations are executed with minimal delay. Additionally, the migration and reproduction phases allow IoT devices to dynamically adjust their task allocations, adapting to network changes and preventing resource overloading. By efficiently balancing the computational load, EWOA significantly prolongs network lifetime, reduces energy dissipation, and ensures seamless execution of IoT tasks while maintaining high system performance.

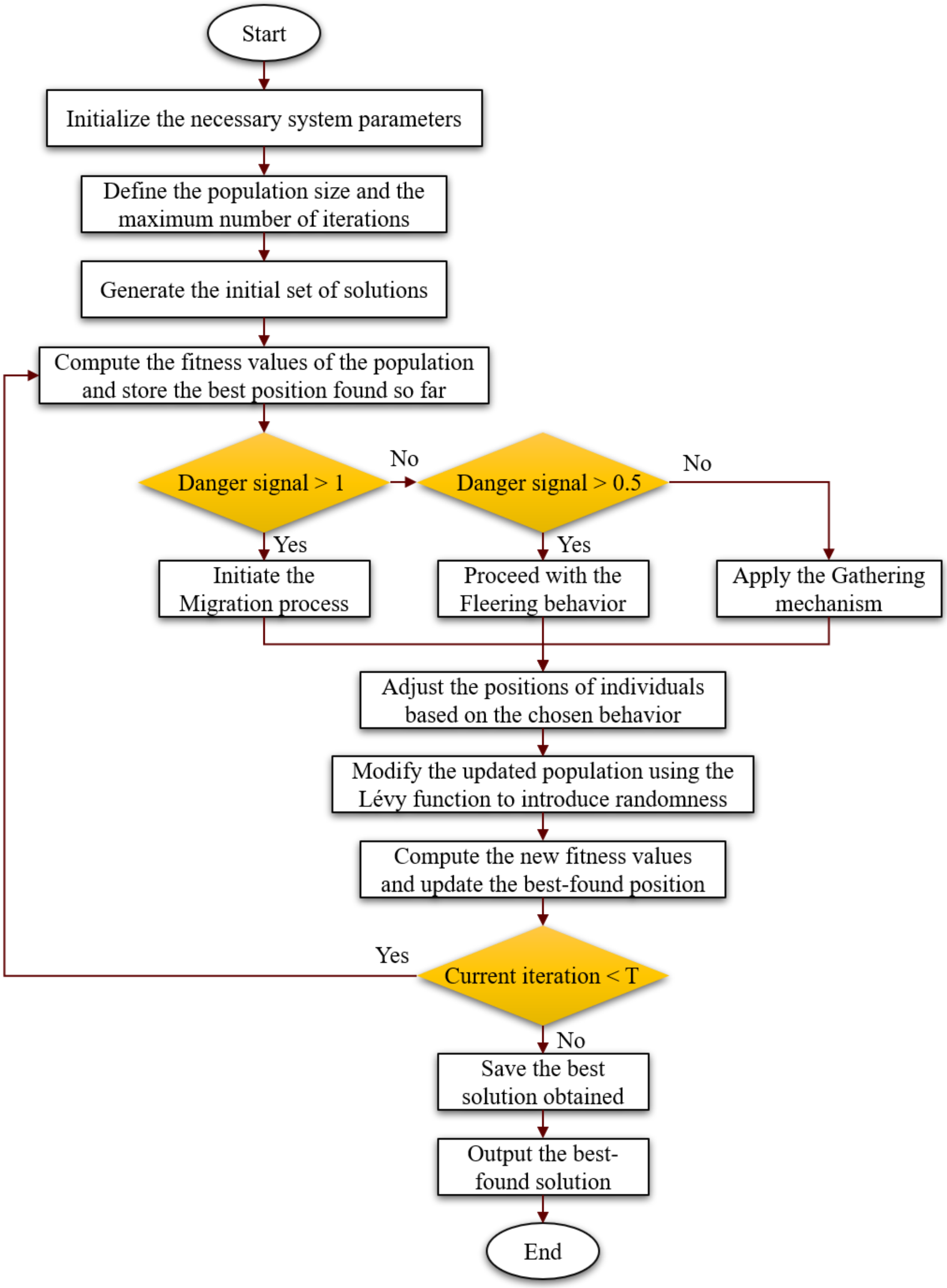


Fig. 3. Flowchart of EWOA.

## V. Results

This section evaluates the proposed EWOA efficiency in scheduling IoT tasks and energy consumption in IoT environments. EWOA efficiency was assessed by comprehensive experiments conducted in MATLAB R2020a, which has high capabilities for matrix computation, model optimization, and effective data manipulation. IoT task scheduling is represented as DAG, with computational nodes arranging tasks under energy and communication costs. Convergence behavior, energy consumption, and cost efficiency performance metrics were analyzed by comparing existing methods, including SA, PSO, and ACO.

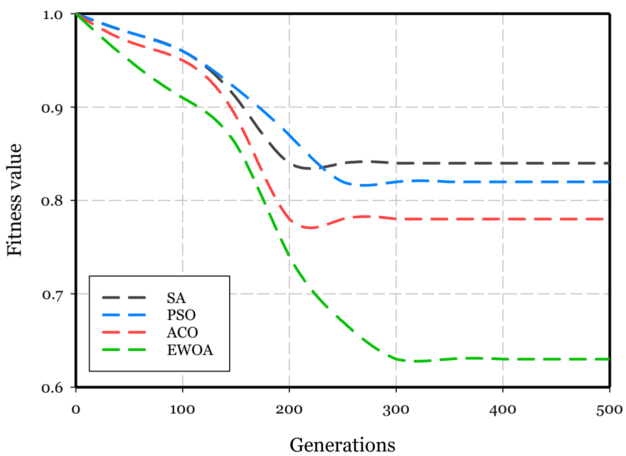


Fig. 4. Fitness value results.

Fig. 4 shows the algorithm behavior for a sequence of several generations. Initially, the fitness value of EWOA was high but continued to fall with a growing number of iterations, reaching equilibrium at approximately the 320th generation. Through refinement over a while, the algorithm avoids early convergence and maintains an optimal balance between search and exploitation. Rapid early improvements confirm that the Lévy flight search mechanism operates effectively. The algorithm can explore a range of solution spaces and settle down to an optimal task distribution scheme.

Efficient energy consumption is one of the most important objectives in IoT task scheduling. Fig. 5 illustrates algorithms' energy consumption with an increased number of tasks. The observations confirm that with an increased number of scheduled tasks, EWOA outperforms all else consistently, consuming much less energy. Simulated Annealing consumes the most energy, with a follow-up consumption in terms of ACO and PSO. That confirms that EWOA maximizes task distribution better, minimizing computational overhead and unnecessary migration of tasks. EWO's use of an adaptive exploration mechanism and task prioritisation techniques are responsible for its high energy efficiency. Therefore, EWO is an ideal selection for IoT large-scale implementations.

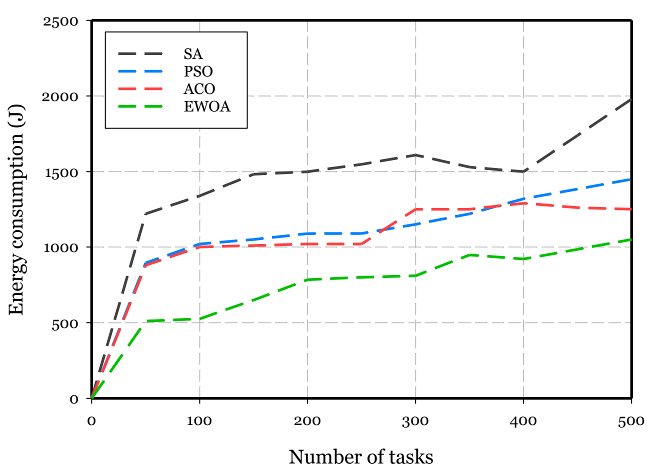


Fig. 5. Energy consumption results.

In addition to efficiency in terms of energy, cost-minimizing is a significant concern in IoT resource management. Fig. 6 is a comparative analysis of cost efficiency in terms of algorithms. EWOA consistently generates lesser values for cost when compared with other algorithms with a larger number of tasks. All methodologies have an increasing trend in terms of cost with growing complexity in terms of tasks. Still, EWOA experienced a moderate and less sharp rise, representing its resource distribution efficiency. Intelligent mapping of tasks with resources and adaptability in EWOA enables it to have a lesser cost, representing its effectiveness in IoT task scheduling with an efficient computational burden at an economical cost.

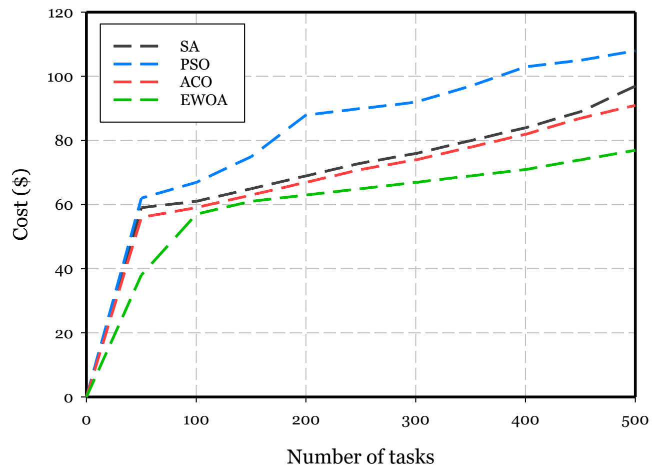


Fig. 6. Cost results.

The results clearly show that EWOA introduces significant improvement over conventional approaches in terms of velocity of convergence, efficiency in terms of energy, and cost savings. Lévy flight, multi-objective estimation of fitness, and flexible distribution of resources techniques included in EWOA make EWOA efficient in working with IoT scenarios with changing environments. High performance in various performance factors proves that EWOA can effectively work for big IoT scenarios with high regard for conserving energy and minimizing cost. With smart optimizing techniques, the proposed algorithm effectively distributes computational loads, maximizes network life, and maximizes effective use of resources, and thus can work as an efficient alternative for future IoT networks.

## VI. Discussion

Evolution towards energy-efficient task schedules in IoT contexts necessitates adaptive yet robust optimization mechanisms. EWOA met this need by providing a new hybrid integration of Lévy flight search with adaptive search methods. This work contributes significantly to current literature in resource-constrained optimization using metaheuristics, ranking EWOA superior in maintaining a balance between exploitation and exploration.

A key contribution of this work is its multi-objective formulation, where energy consumption, execution latency, and resource utilization are simultaneously considered. Most methods in the literature consider energy efficiency a secondary optimization objective [21], while this EWOA makes energy efficiency central within its scheduling objective. This change is crucial in IoT applications where devices rely on batteries operating in remote or hostile regions, and energy exhaustion can impact system reliability and longevity.

As opposed to traditional metaheuristics such as PSO and ACO, which are subject to premature convergence and low adaptability to dynamic workloads [22], EWOA applies Lévy flight to move out of local optima while having search diversity throughout large iterations. This makes EWOA effective even for complex non-linear task dependencies best represented by Directed Acyclic Graph (DAG) structures. These are critical to scalability in smart cities, industrial IoT, and fog/edge computing.

The algorithm's biological inspiration, from walrus migration and reproduction patterns, provides a unique and underexplored behavioral model in the metaheuristic landscape. While swarm intelligence and evolutionary paradigms dominate the literature, the walrus model demonstrates that new behavioral analogies can yield valuable heuristic principles, particularly with advanced statistical tools like Lévy distributions.

EWOA solves several open problems in this area. Firstly, its volatility-adaptive structure enables flexibility, a key necessity for self-organizing fault-tolerant systems. Second, its dynamic allocation facility ensures it is a prime candidate for real-time systems where device status may switch. Third, it provides a generalizable model for hybrid optimization methods. It creates possibilities for intermingling with federated learning agents, machine learning predictors, or context-dependent controllers for smart decision-making in dynamic environments.

Despite this, real-world deployment can be hindered by certain practical limitations, namely computational complexity in large-scale systems and requirements for hyperparameter fine-tuning to prevent poor performance. Although MATLAB simulations assure controlled and reproducible evaluation circumstances, actual tests using physical IoT testbeds or emulator environments would thoroughly validate the algorithm's robustness against network noise, packet loss, or hardware constraints.

## VII. CONCLUSION

This study designed EWOA for optimal use in IoT scheduling. It presented a computation-efficient approach for reducing computational costs and improving system performance using optimal energy and resource usage. EWOA facilitated a dynamic exploitation-exploration tradeoff using Lévy flight-inspired movement strategies, multi-objective optimization, and flexible task allocation for optimal scheduling. By modeling the scheduling problem in a DAG, EWOA effectively distributes tasks regarding computational demand, energy consumption, and communication overhead to IoT nodes.

Simulation tests have proven that EWOA outperforms traditional algorithms. Convergence analysis revealed that EWOA achieves optimal performance at an optimal rate without premature convergence and with an ideal task distribution. Analysis of energy consumption and cost has proven that EWOA minimizes energy loss and computational costs. Intelligent mapping of resources to tasks and a flexible scheduling mechanism ensure a network's durability and service efficiency, and EWOA is a potential real-time IoT candidate.

Although the proposed scheme reflects considerable improvements in terms of energy efficiency and task execution performance, additional enhancements are possible in future research. Machine learning predictive models can improve scheduling efficiency, accuracy, and adaptability in dynamic IoT environments. Implementations in edge and fog architectures can reveal even deeper insights into real-life implementations. Hybrid frameworks combining metaheuristic algorithms with deep learning for even better performance in complex IoT networks can be developed in future studies.

## REFERENCES

[1] Salama, R., F. Al-Turjman, P. Chaudhary, and S.P. Yadav. (Benefits of Internet of Things (IoT) Applications in Health care-An Overview). in 2023 International Conference on Computational Intelligence, Communication Technology and Networking (CICTN). 2023. IEEE.

[2] Padmanaban, S., M.A. Nasab, M. Hatami, O.H. Milani, M.A. Dashtaki, M.A. Nasab, and M. Zand, The Impact of the Internet of Things in the Smart City from the Point of View of Energy Consumption Optimization. Biomass and Solar - Powered Sustainable Digital Cities, 2024: p. 81-122

[3] Pourghebleh, B., N. Hekmati, Z. Davoudnia, and M. Sadeghi, A roadmap towards energy - efficient data fusion methods in the Internet of Things. Concurrency and Computation: Practice and Experience, 2022. 34(15): p. e6959

[4] Kamalov, F., B. Pourghebleh, M. Gheisari, Y. Liu, and S. Moussa, Internet of medical things privacy and security: Challenges, solutions, and future trends from a new perspective. Sustainability, 2023. 15(4): p. 3317

[5] Anvigh, A.A., Y. Khavan, and B. Pourghebleh, Transforming Vehicular Networks: How 6G can Revolutionize Intelligent Transportation? Science, Engineering and Technology, 2024. 4(1): p. 80-93

[6] Ahamad, R. and K.N. Mishra, Hybrid approach for suspicious object surveillance using video clips and UAV images in cloud-IoT-based computing environment. Cluster Computing, 2024. 27(1): p. 761-785

[7] Shah, S.K., K. Joshi, S. Khantwal, Y.S. Bisht, H. Chander, and A. Gupta. IoT and WSN integration for data acquisition and supervisory control. in 2022 IEEE World Conference on Applied Intelligence and Computing (AIC). 2022. IEEE.

[8] Rivandi, E., FinTech and the Level of Its Adoption in Different Countries Around the World. Available at SSRN 5049827, 2024.https://dx.doi.org/10.2139/ssrn.5049827

[9] Rivandi, E. and R. Jamili Oskouie, A Novel Approach for Developing Intrusion Detection Systems in Mobile Social Networks. Available at SSRN 5174811, 2024.https://dx.doi.org/10.2139/ssrn.5174811

[10] Liu, L., H. Zhang, and Y. Liu, A smart and transparent district heating mode based on industrial Internet of things. International Journal of Energy Research, 2021. 45(1): p. 824-840

[11] Zormati, M.A., H. Lakhlef, and S. Ouni, Review and analysis of recent advances in intelligent network softwarization for the Internet of Things. Computer Networks, 2024: p. 110215

[12] Shen, Y., X. Zhu, Z. Guo, K. Yu, O. Alfarraj, V.C. Leung, and J.J. Rodrigues, A Deep Learning-Based Data Management Scheme for Intelligent Control of Wastewater Treatment Processes Under Resource-Constrained IoT Systems. IEEE Internet of Things Journal, 2024

[13] Hosseinzadeh, A., M. Shahin, M. Maghanaki, H. Mehrzadi, and F.F. Chen, Minimizing wastevia novel fuzzy hybrid stacked ensembleof vision transformers and CNNs to detect defects in metal surfaces. The International Journal of Advanced Manufacturing Technology, 2024: p. 1-26.10.1007/s00170-024-14741-y

[14] Mahdimahalleh, S.E., Revolutionizing wireless networks with federated learning: A comprehensive review. arXiv preprint arXiv:2308.04404, 2023.https://doi.org/10.48550/arXiv.2308.04404

[15] Wadhwa, H. and R. Aron, Optimized task scheduling and preemption for distributed resource management in fog-assisted IoT environment. The Journal of Supercomputing, 2023. 79(2): p. 2212-2250

[16] Kermani, A., A.M. Jamshidi, Z. Mahdavi, A.a. Dashtaki, M. Zand, M.A. Nasab, T. Samavat, P. Sanjeevikumar, and B. Khan, Energy management system for smart grid in the presence of energy storage and photovoltaic systems. International Journal of Photoenergy, 2023. 2023(1): p. 5749756.https://doi.org/10.1155/2023/5749756

[17] Bu, T., Z. Huang, K. Zhang, Y. Wang, H. Song, J. Zhou, Z. Ren, and S. Liu, Task scheduling in the internet of things: challenges, solutions, and future trends. Cluster Computing, 2024. 27(1): p. 1017-1046

[18] Azizi, S., M. Shojafar, J. Abawajy, and R. Buyya, Deadline-aware and energy-efficient IoT task scheduling in fog computing systems: A semi-greedy approach. Journal of network and computer applications, 2022. 201: p. 103333

[19] Wang, H., Y. Wang, X. Xie, and M. Li, A scheduling scheme for minimizing age under delay tolerance in IoT systems with heterogeneous traffic. IEEE Internet of Things Journal, 2024

[20] Khiabani, P.M., G. Danala, W. Jentner, and D. Ebert. Challenges in Data Integration, Monitoring, and Exploration of Methane Emissions: The Role of Data Analysis and Visualization. in 2024 IEEE Workshop on Energy Data Visualization (EnergyVis). 2024. IEEE.

[21] Weikert, D., C. Steup, and S. Mostaghim, Availability-aware multi-objective task allocation algorithm for internet of things networks. IEEE Internet of Things Journal, 2022. 9(15): p. 12945-12953

[22] Ren, X., Z. Zhang, S. Chen, and K. Abnoosian, An energy - aware method for task allocation in the Internet of things using a hybrid optimization algorithm. Concurrency and Computation: Practice and Experience, 2021. 33(6): p. e5967

[23] Bali, M.S., R. Alroobaea, S. Algarni, M. Alsafyani, K. Mohiuddin, K. Gupta, and D. Gupta, An efficient task allocation framework for scheduled data in edge based Internet of Things using hybrid optimization algorithm approach. Physical Communication, 2023. 58: p. 102047

[24] Nematollahi, M., A. Ghaffari, and A. Mirzaei, Task and resource allocation in the internet of things based on an improved version of the moth-flame optimization algorithm. Cluster Computing, 2024. 27(2): p. 1775-1797

[25] Nematollahi, M., A. Ghaffari, and A. Mirzaei, Task offloading in Internet of Things based on the improved multi-objective aquila optimizer. Signal, Image and Video Processing, 2024. 18(1): p. 545-552

[26] Satouf, A., A. Hamidoğlu, Ö.M. Gül, A. Kuusik, L. Durak Ata, and S. Kadry, Metaheuristic-based task scheduling for latency-sensitive IoT applications in edge computing. Cluster Computing, 2025. 28(2): p. 1-17

[27] Umer, A., M. Ali, A.I. Jehangiri, M. Bilal, and J. Shuja, Multi-objective task-aware offloading and scheduling framework for internet of things logistics. Sensors, 2024. 24(8): p. 2381

[28] Trojovský, P. and M. Dehghani, Walrus optimization algorithm: a new bio-inspired metaheuristic algorithm. 2022