Journal of Information Systems Engineering and Management
2025, Vol(59s)
e-ISSN: 2468-4376
https://jisem-journal.com/
Research Article

A Multi-Objective Genetic Algorithm Optimization of Delay and UAV Energy Consumption for Task Offloading in Flying Fog Computing

Abdessamed Sassi[1], Mohamed Amine ATTALAH[2], Sofiane ZAIDI[3], Carlos T. CALAFATE[4], Nour El Houda Dehimi[1]
[1]*Department of Mathematics and Computer Science, Research Laboratory of Artificial Intelligence and Autonomous Objects, University of Oum El Bouaghi, Oum El Bouaghi 04000, Algeria*
[2]*Department of Electronics, University Center of Tipaza, Tipaza 42000, Algeria*
[3]*Department of Mathematics and Computer Science, Research Laboratory on Computer Science's Complex Systems (RELA(CS)2), University of Oum El Bouaghi, Oum El Bouaghi 04000, Algeria*
[4]*Department of Computer Engineering (DISCA), Universitat Politècnica de València, 46022 Valencia, Spain*

ARTICLE INFO

Received: 01 Aug 2025
Revised: 12 Sept 2025
Accepted: 22 Sep 2025

ABSTRACT

This paper addresses the challenge of efficient task offloading in Fog Computing for Internet of Drones (IoD) applications by introducing a multi-objective optimization framework. Unlike previous studies that optimize either delay or energy consumption in isolation, our approach jointly considers both metrics through a hybrid architecture that combines Fog nodes (UAVs and base stations) with Cloud resources. We propose and evaluate three multi-objective metaheuristic algorithms – Multi-Objective Particle Swarm Optimization (MOPSO), Multi-Objective Genetic Algorithm (MOGA), and Multi-Objective Ant Colony Optimization (MOACO) – to enhance offloading efficiency. Simulation results show that all three methods improve latency and UAV energy efficiency; however, MOGA consistently achieves the best overall performance in high-resource configurations. These results demonstrate MOGA's effectiveness in managing the trade-off between offloading delay and energy consumption in dynamic UAV-based networks, confirming its potential for scalable and energy-efficient task offloading in future IoD-Fog Computing environments.

Keywords: Internet of flying things, task offloading, Fog Computing, energy consumption, unmanned aerial vehicles.

I. INTRODUCTION

Unmanned Aerial Vehicles (UAVs), or drones, are becoming one of the most important technologies in various fields, as they are widely available, smaller in size, and less expensive than ever before. They now play vital roles in diverse applications, including delivery services, surveillance, search-and-rescue operations, and environmental monitoring **[1-3]**. Moreover, UAVs can reduce costs and risks, accelerate some tasks, and reach places that people or regular vehicles cannot easily access **[4]**. However, UAVs cannot handle computationally intensive tasks (e.g., those requiring a lot of memory and processing power), and therefore need additional Computing resources **[5]**. Due to the limited onboard resources and Computing capabilities of UAVs, processing their data locally is a challenging task **[6]**. Consequently, certain Computing tasks need to be offloaded from UAVs to either Fog base stations (Fog BSs), or Fog UAVs within the network for remote processing **[4]**, thereby forming what is known as the Internet of Flying Things (IoFT) **[7]**. Both Fog BSs and Fog UAVs act as mobile and distributed nodes, providing computation, storage, and networking services **[8-11]**. Within Flying Fog Computing environments, task offloading plays a vital role in overcoming the computational limitations of UAVs **[12, 13]**. As UAVs are constrained by limited onboard resources and battery life, it is inefficient – or sometimes infeasible – for them to execute all computational tasks locally. Task offloading lets UAVs pass heavy or urgent tasks to more powerful Fog nodes, depending on resource availability,

Journal of Information Systems Engineering and Management
2025, Vol(59s)
e-ISSN: 2468-4376
https://jisem-journal.com/ **Research Article**

energy limits, and network conditions. This helps reduce delays, share the work more evenly, and extend UAV flight time by saving onboard energy. Optimizing task offloading is key to making Flying Fog Computing work well for fast-changing, mobile, and time-sensitive IoT applications.

Despite its benefits, task offloading in Flying Fog Computing faces significant challenges. Given UAVs' limited battery life and Computing power, smart energy-aware offloading strategies are required. Task offloading can conserve energy by sending computations to other nodes, but it also consumes power during data transmission. Energy-aware strategies must balance these trade-offs to extend UAV life and improve system reliability in dynamic IoT environments. Energy management is vital for UAVs in Flying Fog Computing, as power is consumed by flight, processing data, and communication. Poor energy control can cause mission failure or data loss, especially in emergencies.

While several studies in Flying Fog Computing employ metaheuristic algorithms for task offloading, they typically optimize a single objective—either minimizing task execution delay or reducing UAV energy consumption—without considering both together. In contrast, our work addresses these two objectives simultaneously through a multi-objective optimization framework. Specifically, we propose and compare three metaheuristic algorithms— Multi-Objective Genetic Algorithm (MOGA), Multi-Objective Particle Swarm Optimization (MOPSO), and Multi-Objective Ant Colony Optimization (MOACO)—that jointly optimize task delay and UAV energy consumption within a hybrid Fog architecture combining stationary and mobile Fog nodes. Unlike earlier GA-based and PSO-based approaches, our framework dynamically adapts to jointly minimize delay and energy consumption in UAV environments. Simulation results show that MOGA consistently outperforms the other approaches, providing a robust solution for task offloading in Flying Fog Computing.

The remainder of this paper is organized as follows: Section II discusses the related work. Section III describes the proposed system architecture, formulates the task offloading problem, and introduces the multi-objective optimization algorithms: MOGA, MOPSO, and MOACO. Section IV presents the simulation setup and evaluates their performance. Finally, Section V concludes the paper and highlights potential directions for future research.

II. RELATED WORK

In recent years, many studies have investigated intelligent approaches to task sharing and resource management in UAV-assisted Edge Computing. These methods aim to handle the growing need for fast responses and heavy data processing in network services. The authors in **[12]** proposed PSO BS-Fog, an optimization approach for task offloading that integrates the PSO heuristic with Fog Computing technology in the IoD. The proposed solution uses PSO to offload tasks from UAVs to Fog base stations, aiming to reduce offloading delays and increase available storage and processing capacity. However, it does not explicitly address UAV battery consumption, which remains a critical limitation in energy-constrained UAV environments. The authors in **[13]** introduced GA Hybrid-Fog, a task-offloading optimization strategy that uses a heuristic genetic algorithm (GA) combined with hybrid Fog Computing for the Internet of Drones. The solution reduces offloading delays by offloading tasks from Edge UAVs to both Fog BSs and Fog UAVs, thereby enhancing processing and storage capacity. Nevertheless, this approach mainly focuses on delay reduction and does not jointly optimize energy consumption, which limits its suitability for dynamic, energy-sensitive UAV scenarios.

Other researchers have applied deep reinforcement learning (DRL) to manage and control unmanned aerial vehicles (UAVs) **[14]**. The authors in **[15]** presented a multi-agent deep reinforcement learning framework that combines the Distance to Task Location and Capability Match (DTLCM) method with the Multi-Agent Deep Deterministic Policy Gradient (MADDPG) algorithm. The proposed framework jointly improves UAV path planning, task offloading, resource utilization, and communication management. The optimization uses both the UAVs' processing ability and their distance from tasks to improve energy use and shorten delays. The authors in **[16]** aimed to jointly optimize trajectory planning, task offloading, and resource allocation (TTR). They proposed a hybrid solution consisting of a PSO-based algorithm (P-TTR) and a deep reinforcement learning method (S-TTR). In **[17]**, a UAV-assisted system combining Cloud and Edge Computing is proposed for 5G networks. Using a DDPG algorithm to optimize scheduling, trajectory, and offloading, the approach significantly reduces delays, with hybrid Cloud–Edge offloading performing best. In **[18]**, the authors propose **TinyDeepUAV**, a multi-objective DRL framework based on TinyML for resource-

Journal of Information Systems Engineering and Management
2025, Vol(59s)
e-ISSN: 2468-4376
https://jisem-journal.com/ **Research Article**

constrained UAV–IoT systems. By using vector-based reinforcement and D3QN, it balances delay and energy consumption with low-complexity offloading decisions. In **[19]**, the authors proposed a novel framework combining DRL with Zero-Reflection Intelligent Surfaces (Ze-RIS) to optimize task offloading in UAV-Mobile Edge Computing (UAV-MEC) systems. In **[20]**, the authors proposed an iterative approach that optimizes task offloading, data transmission, UAV computational capacity, UAV positioning, and service latency. In **[21]**, the authors proposed an optimization framework for task offloading in a collaborative UAV environment integrated with Fog and Cloud Computing resources. The authors modelled the task-offloading problem as a Mixed-Integer Linear Programming (MILP) problem to jointly optimize UAV energy use and offloading latency.

Overall, existing studies in Flying Fog Computing mainly concentrate on optimizing task offloading on either delay or UAV energy consumption, but rarely address both simultaneously. In contrast, our work considers these two objectives together within a unified multi-objective optimization framework. To this end, we propose and compare three metaheuristic approaches—MOPSO, MOGA, and MOACO—standing for multi-objective Particle Swarm Optimization, Genetic Algorithm, and Ant Colony Optimization, respectively.

III. PROPOSED APPROACH

We propose an architecture for task offloading from Edge UAVs to Fog UAVs and Fog BSs, aiming to optimize both total task offloading delay and UAVs energy consumption using the following proposed methods: MOPSO, MOGA and MOACO. In this architecture, three types of nodes are involved: Edge UAVs, Fog UAVs, and Fog BSs (see Fig.1).

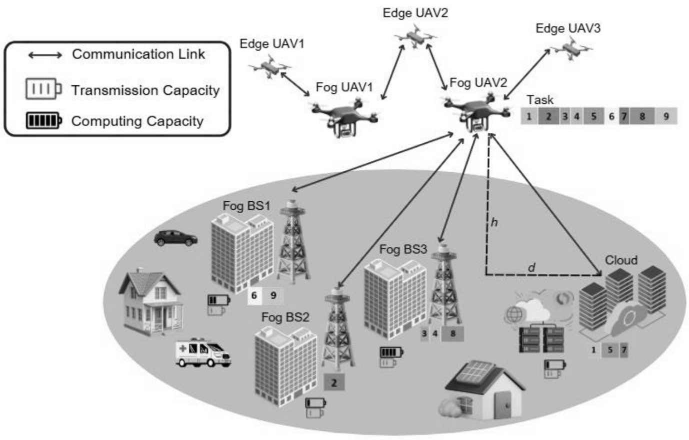

Figure 1 Architecture of the proposed model.

3.1 Mathematical model

Fig. 1 presents the architecture of the proposed model. To model channel connectivity across large geographical areas, it is necessary to compute the transmission delay between aerial nodes (UAVs or Fog UAVs) and ground devices (Fog BSs). Ensuring seamless data transmission between these nodes requires maintaining adequate network quality, which is closely related to the transmission delay. This delay can be expressed in terms of the transmission rate R_i, as shown in Eq. (1) [10]:

$$D_{ij} = \frac{S_j}{R_i} \tag{1}$$

Where S_j denotes the data size of task j. The transmission rate R_i depends on the channel conditions and is determined using Eq. (2), Eq. (3), and Eq. (4). Based on the transmission bandwidth W, the maximum achievable data rate from an aerial to a ground device is defined by [11]:

$$R_i = w \cdot \ log_2(1 + SNR_i) \tag{2}$$

The signal-to-noise ratio SNR_i is calculated as:

$$SNR_i = \frac{P_{tx}}{L_{dB}\,(d_i) \cdot N_0} \tag{3}$$

Journal of Information Systems Engineering and Management
2025, Vol(59s)
e-ISSN: 2468-4376
https://jisem-journal.com/ **Research Article**

Where P_{tx} is the transmit power, $L_{dB}(d_i)$ is the path loss in decibels at distance d_i, and N_0 is the noise power. The path loss $L_{dB}(d_i)$ is calculated using the free-space path loss model:

$$L_{dB}(d_i) = 20\log(d_i) + 20\log(f) + 20\log\left(\frac{4\pi}{c}\right) + L_{ex} \quad (4)$$

Here, d_i is the distance between the aerial node UAV_i and the corresponding ground device Fog BS_i (as depicted in Fig. 1), f is the carrier frequency, c is the speed of light, and L_{ex} represents the excess path loss, accounting for both Line-of-Sight (LoS) and Non-Line of Sight (NLoS) components [10].

3.1.1 Computing model

The Computing time of the Fog node i depends on its CPU frequency f_i. The CPU cycles required to process each bit of data are represented by η_j. The Computing delay of task j in Fog node i, denoted as C_{ij}, is calculated using [11]:

$$C_{ij} = \frac{S_j \cdot \eta_j}{f_i} \quad (5)$$

3.1.2 Energy consumption model

The total energy consumption includes both transmission and processing energy. The transmission energy E_{ij}^{tx} required by the UAV i to send task j is given by:

$$E_{ij}^{tx} = P_{tx} \cdot D_{ij} = P_{tx} \cdot \frac{S_j}{R_i} \quad (6)$$

The processing energy E_{ij}^{proc} to execute task j on Fog node i is calculated as:

$$E_{ij}^{proc} = \kappa \cdot S_j \cdot \eta_i \cdot f_i^2 \quad (7)$$

Where: κ is the effective switched capacitance.

3.1.3 Offloading cost

To make an optimal offloading decision, a cost function is defined to capture the trade-off between task latency and energy consumption. The offloading cost $Cost_{ij}$ reflects this balance when task j is processed by Fog node i, and is defined as:

$$Cost_{ij} = \alpha_i \cdot D_{ij} + \beta_i \cdot \left(E_{ij}^{tx} + E_{ij}^{proc}\right) \quad (8)$$

Where :

- α_i and β_i are weighting coefficients such that $\alpha_i + \beta_i = 1$
- D_{ij} represents the total delay for executing task j on node i (see Eq. (1)),
- E_{ij}^{tx} is the energy consumed for transmitting the task to node i (see Eq. (6)),
- E_{ij}^{proc} is the energy required to process the task at node i (see Eq. (7)).

To account for the energy constraints of UAVs, the weighting factors α_i and β_i are dynamically adjusted based on the current battery level $B_i \in [0, 1]$ of UAV i:

$$\alpha_i = B_i, \quad \beta_i = 1 - B_i \quad (9)$$

This ensures that UAVs with lower battery levels prioritize energy saving, while those with higher battery levels prioritize minimizing delay.

3.2 Algorithmic structure

3.3.1 Task offloading at Edge UAV

The task offloading strategy adopted by the Edge UAV is based on the size of the incoming task. This strategy is proceeded as follows:

Journal of Information Systems Engineering and Management
2025, Vol(59s)
e-ISSN: 2468-4376
https://jisem-journal.com/ Research Article

- **Small tasks** (*Task_size* < *TSMin*): Executed locally on the Edge UAV in order to reduce latency and avoid communication overhead.
- **Medium tasks** (*TSMin* ≤ *Task_size* < *TSMax*): Offloaded to either a Fog UAV or a Fog BS, based on a Multi-Objective Optimization (MOO) decision mechanism.
- **Large tasks** (Task_size ≥ TSMax): Offloaded directly to the Cloud, which provides high computational capacity and storage resources.

3.3.2 Delay, and energy consumption calculation algorithms

In the proposed architecture, multiple Edge UAVs can offload their tasks to various Fog UAVs, Fog Base Stations (Fog BSs), or directly to the Cloud. The distribution of these tasks is optimized to achieve an effective trade-off between offloading delay and UAV energy consumption. This optimization is performed using one of three multi-objective metaheuristic methods: MOGA, MOPSO, or MOACO. The following of this section presents the algorithmic structure of our approach.

In Step 1, the algorithm calculates the data rate based on Eq. (2), Eq. (3) and Eq. (4). In Step 2, it computes both the transmission delay and the processing (Computing) delay for each offloaded task using Eq. (1) and Eq. (5). Finally, the algorithm determines the total offloading delay, which corresponds to the maximum delay among all individual task offloading delays. The total energy consists of two main components: transmission energy and processing energy.

3.3.3 Multi-objective GA algorithm for task offloading in flying Fog Computing

The proposed multi-objective GA algorithm for task offloading in flying Fog Computing, is executed by the Edge UAV to offload tasks to a set of Fog UAVs and Fog BSs. To reduce the transmission delay, the execution time and the consumption energy, the algorithm applies the MOGA operation for a number of iterations. The process begins with Step 1, where the problem is defined by establishing a fitness function. This function evaluates the total offloading delay and total energy consumption for each Fog BS and Fog UAV. In Step 2, each individual in the population is encoded as a binary chromosome, representing a specific allocation of tasks to Fog BSs or Fog UAVs. Then, in Step 3, an SP vector is initialized, where each individual is assigned a fitness value computed using the fitness function.
Step 4, a subset of individuals with the highest fitness values is selected from the population.
Step 5, the GA heuristic applies the crossover function as follows:

- **Step 5.1:** Select two chromosomes j and $j + 1$, if the generated random value r is less than the crossover probability (PC).
- **Step 5.2:** Randomly determine a crossover point that indicates the position of the bits to be exchanged between chromosomes j and $j+1$.
- **Step 5.3:** Perform crossover by exchanging the selected bits of the two chromosomes.

Step 6, the GA heuristic applies the mutation function as follows:

- **Step 6.1:** Select a chromosome j if the regenerated random value r is less than the mutation probability (PM).
- **Step 6.2:** Randomly determine a mutation point that indicates the position of the bit to be mutated in chromosome j.
- **Step 6.3:** Modify the value of the selected bit of chromosome j.

Steps 7 and 8 evaluate the population to obtain a minimal compromise between delay and energy consumption for all task offloading decisions to Fog BSs and Fog UAVs. Finally, Step 9 generates the overall optimal compromise solution in terms of delay and energy consumption for all offloaded tasks.

Journal of Information Systems Engineering and Management
2025, Vol(59s)
e-ISSN: 2468-4376
https://jisem-journal.com/ **Research Article**

IV. EXPERIMENTAL RESULTS

To evaluate the performance of the proposed task offloading strategy, a set of simulations was conducted using MATLAB r2016a ToolBox. In this study, we compare the performance of three multi-objective optimization algorithms: MOGA, MOPSO, and MOACO. These algorithms are evaluated across the simulation scenarios defined in Table 1, with the objective of optimizing task offloading decisions. The comparison focuses on key performance metrics, including total latency and energy consumption. By examining how each algorithm responds to variations in task load, resource availability, and network conditions, we aim to assess their strengths, limitations, and overall suitability for real-time, resource-constrained systems.

Table 1 outlines the set of simulation scenarios used to analyze the behavior of the proposed MOGA algorithm for task offloading under different system conditions. Each scenario varies a specific parameter to observe its individual impact on the performance of the proposed algorithm. These scenarios are designed to test the efficiency and adaptability of the MOGA algorithm. By systematically adjusting factors such as the number of tasks, the number of Fog UAVs, the number of Fog BSs, and channel conditions, the study identifies key elements that influence the scalability and robustness of the offloading strategies.

Table 1. Simulation scenarios and expected insights.

Scenario	Varied Parameter	Expected Insight
A	Number of Edge UAVs (10–100)	Algorithm capacity and manage overload
B	Number of Tasks (10–100)	Algorithm scalability
C	Number of Fog-UAVs	Resource availability effect in the UAV layer
D	Number of Fog-BSs	infrastructure density affects
E	Channel Conditions (Data rate)	Network impact

Fig. 2 illustrates the impact of the number of Edge UAVs on the performance of multi-objective task-offloading methods, comparing MOGA, MOACO, and MOPSO. In this experiment, the number of offloaded tasks is fixed at 100, the number of Fog UAVs is fixed at 10, and the number of Fog BSs is fixed at 10. As shown in the figure, increasing the number of Edge UAVs leads to an increase in the best cost for all optimization algorithms, primarily due to the additional tasks generated by the larger number of Edge UAVs. Furthermore, the results show that the proposed MOGA algorithm achieves a lower best cost, indicating a good trade-off between delay and energy consumption compared to MOACO and MOPSO, as its metaheuristic search yields an optimal task-offloading strategy that minimizes transmission delay, processing delay, and energy consumption.

Fig. 3 illustrates the best cost associated with task offloading for the three algorithms, as a function of the number of tasks. In this experiment, the number of offloaded tasks varies from 50 to 450, while the number of Fog UAVs is fixed at 10, and the number of Fog BSs is fixed at 10. As the number of tasks increases, the best cost of all methods increases due to the greater processing capacity required to handle the additional tasks. Furthermore, the results show that the proposed MOGA algorithm consistently achieves a lower best cost compared to MOACO and MOPSO. This improvement is attributed to the genetic algorithm's ability to heuristically determine an optimal trade-off in task offloading decisions, resulting in minimal delay and reduced energy consumption for task transmission and processing. Fig. 4 and Fig. 5 present, respectively, the impact of the number of Fog UAVs and the number of Fog BSs on the performance of multi-objective task-offloading methods. In both cases, as the number of Fog nodes increases, the best cost of all algorithms decreases, resulting in the higher processing capacity available in the network.

Journal of Information Systems Engineering and Management
2025, Vol(59s)
e-ISSN: 2468-4376
https://jisem-journal.com/ **Research Article**

Moreover, the proposed MOGA algorithm consistently outperforms MOACO and MOPSO in both scenarios.

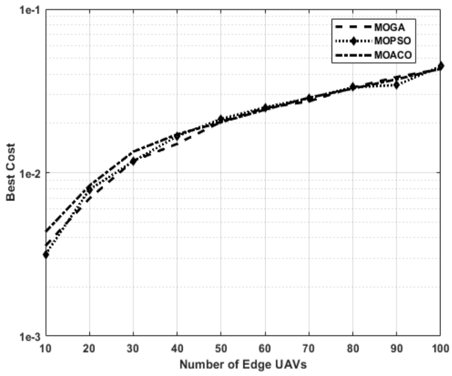

Figure 2. Impact of Edge UAVs quantity on the performance of multi-objective task offloading methods

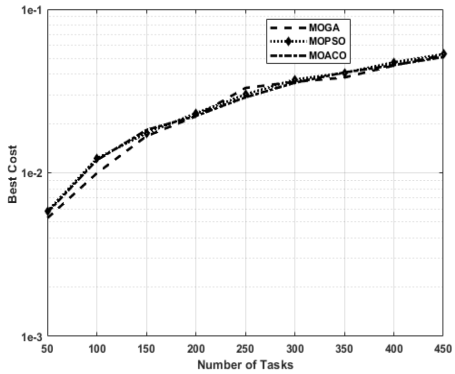

Figure 3. Impact of number of tasks on the performance of multi-objective task offloading methods.

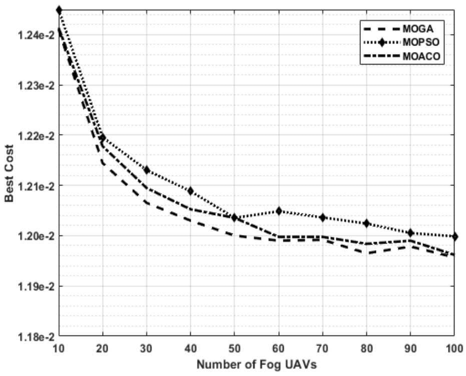

Figure 4. Impact of Fog UAVs quantity on the performance of multi-objective task offloading methods

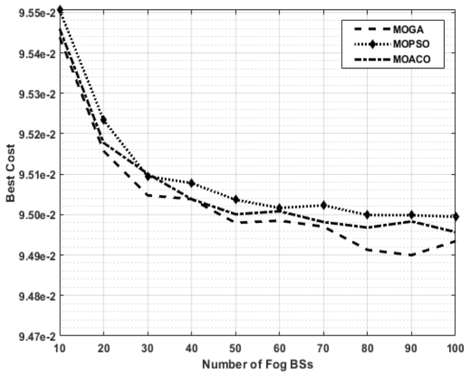

Figure 5. Impact of Fog BSs quantity on the performance of multi-objective task offloading methods.

V. CONCLUSION

In this paper, we have proposed a MOGA-based multi-objective optimization approach for task offloading in Fog Computing environments. The proposed solution leverages the metaheuristic search capability of the Genetic Algorithm to optimize task allocation between Edge UAVs, Fog UAVs, Fog Base Stations, and Cloud, aiming to minimize the best cost, which represents a trade-off between transmission delay, processing delay, and energy consumption. Simulation results showed that MOGA consistently outperforms traditional multi-objective optimization algorithms such as MOACO and MOPSO in various network scenarios. Specifically, MOGA achieved superior performance when varying the number of Edge UAVs, the number of tasks, the number of Fog UAVs, the number of Fog BSs, and the channel conditions. The improvements were particularly evident in high-resource scenarios. The gain is attributed to MOGA's ability to heuristically determine optimal task-offloading strategies under dynamic network conditions. As a future research direction, we intend to extend this work by integrating MOGA with other heuristic or machine learning-based approaches to develop a hybrid offloading framework, capable of adapting to even more complex and large-scale IoD–Fog networks.

Journal of Information Systems Engineering and Management
2025, Vol(59s)
e-ISSN: 2468-4376
https://jisem-journal.com/ **Research Article**

REFRENCES

[1] M. Hassanalian and A. Abdelkefi, “Classifications, applications, and design challenges of drones: A review,” *Prog. Aerosp. Sci.*, vol. 91, pp. 99–131, May 2017, doi: 10.1016/j.paerosci.2017.04.003.
[2] W. Y. H. Adoni, J. S. Fareedh, S. Lorenz, R. Gloaguen, Y. Madriz, A. Singh, and T. D. Kühne, “Intelligent swarm: Concept, design and validation of self-organized UAVs based on leader–followers paradigm for autonomous mission planning,” *Drones*, vol. 8, no. 10, p. 575, 2024.
[3] S. Chen, W. Li, W. Zheng, F. Liu, S. Zhou, S. Wang, Y. Yuan, and T. Zhang, “Application of optical communication technology for UAV swarm,” *Electronics*, vol. 14, no. 5, p. 994, Feb. 2025.
[4] J. Almutairi, M. Aldossary, H. A. Alharbi, B. A. Yosuf, and J. M. H. Elmirghani, “Delay-optimal task offloading for UAVenabled Edge-Cloud Computing systems,” *IEEE Access*, vol. 10, pp. 51575–51586, 2022.
[5] D. Rahbari, M. M. Alam, Y. L. Moullec, and M. Jenihhin, “Fast and fair computation offloading management in a swarm of drones using a rating-based federated learning approach,” *IEEE*, vol. 9, pp. 113832–113849, 2021.
[6] N. Mohamed, J. Al-Jaroodi, I. Jawhar, H. Noura, and S. Mahmoud, “UAVFog: A UAV-based Fog Computing for Internet of Things,” in *Proc. IEEE SmartWorld, Ubiquitous Intell. Comput., Adv. Trusted Comput., Scalable Comput. Commun., Cloud Big Data Comput., Internet People Smart City Innov*, Aug. 2017, pp. 1–8.
[7] F. H. Rahman, S. H. S. Newaz, T.-W. Au, W. S. Suhaili, M. A. P. Mahmud, and G. M. Lee, “EnTruVe: ENergy and TRUst-aware virtual machine allocation in vehicle Fog Computing for catering applications in 5G,” *Future Gener. Comput. Syst.*, vol. 126, pp. 196–210, Jan. 2022, doi:10.1016/j.future.2021.07.036.
[8] S. Zaidi, M. Atiquzzaman, and C. T. Calafate, “Internet of flying things (IoFT): A survey,” *Comput. Commun.*, vol. 165, pp. 53–74, Jan. 2021, doi: 10.1016/j.comcom.2020.10.018.
[9] M. Ogab, S. Zaidi, A. Bourouis, and C. T. Calafate, “Machine Learning-Based Intrusion Detection Systems for the Internet of Drones: A Systematic Literature Review,” *IEEE Access.*, Mai.2025.
[10] M. Ogab, S. Zaidi, A. Bourouis, et al., “Machine learning-based intrusion detection systems for the Internet of Drones: A systematic mapping study,” in Proc. 3rd Int. Conf. Comput. Sci. Complex Syst. Their Appl. (ICCSA’24), Oum El Bouaghi, Algeria, Nov. 13–14, 2024.
[11] R. Bouzoubia, S. Zaidi, L. Khamer, A. Bourouis, and M. Ogab, “Deep reinforcement learning-based resource allocation for the Internet of Vehicles: A systematic mapping study,” in Proc. 3rd Int. Conf. Comput. Sci. Complex Syst. Their Appl. (ICCSA’24), Oum El Bouaghi, Algeria, Nov. 13–14, 2024.
[12] S. Zaidi, M. A. Attalah, L. Khamer, *et al.*, “Task offloading optimization using PSO in Fog Computing for the Internet of Drones,” *Drones*, vol. 9, no. 1, p. 23, Jan. 2024, doi: 10.3390/drones9010023.
[13] M. A. Attalah, S. Zaidi, N. Mellal, *et al.*, “Task-offloading optimization using a genetic algorithm in hybrid Fog Computing for the Internet of Drones,” *Sensors*, vol. 25, no. 5, p. 1383, Mar. 2025, doi: 10.3390/s25051383.
[14] A. T. Azar, A. Koubaa, N. A. Mohamed, H. A. Ibrahim, Z. F. Ibrahim, M. Kazim, A. Ammar, B. Benjdira, A. M. Khamis, I. A. Hameed, and G. Casalino, “Drone deep reinforcement learning: A review,” *Electronics*, vol. 10, no. 9, p. 999, May 2021, doi: 10.3390/electronics10090999.
[15] S. A. Akaryia, M. Meaad, T. Nabil, et al., “Task offloading and resource allocation for multi-UAV asset Edge Computing with multi-agent deep reinforcement learning,” *Computing*, vol. 107, no. 5, pp. 1–31, 2025.
[16] S. Akter, D. V. A. Duong, and S. Yoon, “Joint optimization of UAV trajectory, task offloading, and resource allocation in UAV-aided emergency response operations,” *IEEE Internet of Things Journal*, Jan. 2025.
[17] H. Chen, H. Cui, J. Wang, et al., “Computation offloading optimization for UAV-based Cloud-Edge collaborative task scheduling strategy,” *IEEE Transactions on Cognitive Communications and Networking*, 2025.
[18] S. Bebortta, S. S. Tripathy, S. B. Khan, M. M. Al Dabel, A. Almusharraf, and A. K. Bashir, “TinyDeepUAV: A Tiny Deep Reinforcement Learning Framework for UAV Task Offloading in Edge-Based Consumer Electronics,” *IEEE Transactions on Consumer Electronics*, vol. 70, no. 4, pp. 7357–7364, Nov. 2024.
[19] M. N. Tariq, J. Wang, S. Memon, M. Siraj, M. Altamimi, and M. A. Mirza, “Toward Energy-Efficiency: Integrating DRL and ze-RIS for Task Offloading in UAV-MEC Environments,” *IEEE Access*, vol. 12, pp. 65530–65542, 2024, doi: 10.1109/ACCESS.2024.3397890.
[20] Y. Zhou, C. Pan, P. L. Yeoh, K. Wang, M. Elkashlan, B. Vucetic, and Y. Li, “Secure communications for UAV-enabled mobile Edge Computing systems,” *IEEE Transactions on Communications*, vol. 68, no. 1, pp. 376–388, Jan. 2020, doi: 10.1109/TCOMM.2019.2947921.

Journal of Information Systems Engineering and Management
2025, Vol(59s)
e-ISSN: 2468-4376
https://jisem-journal.com/

Research Article

[21] M. Aldossary, "Optimizing Task Offloading for Collaborative Unmanned Aerial Vehicles (UAVs) in Fog-Cloud Computing Environments," *IEEE Access*, vol. 12, pp. 74698–74710, 2024.