International Journal of Applied Operational Research
Vol. 13, No. 4, pp. 31-41, Autumn 2025

Journal homepage: ijorlu.liau.ac.ir

An optimal hybrid firefly optimization and Min-Min algorithm for dynamic resource management in fog nodes for IoT networks

V. Mokhtari, N. Mikaeilvand*, A. Mirzaei, B. Nouri-Moghaddam, S. Jahanbakhsh Gudakahriz

Received: 20 January 2025 ; **Accepted:** 26 March 2025

Abstract With the proliferation of Internet of Things (IoT) devices, predicted to reach 75 billion by 2025, dynamic resource management in Fog Nodes faces challenges in delay-sensitive applications such as smart cities and autonomous vehicles. This paper presents a hybrid framework that integrates the Firefly Optimization Algorithm (FFO) for Virtual Machine (VM) migration and the Min-Min algorithm for rapid convergence probability with fog computations. Instead of generating a completely random population, a chromosome is generated using the greedy Min-Min algorithm, providing a high-quality initial solution, while FFO optimizes resource allocation and VM migration. Simulations on the dataset showed that the proposed method - based on the Min-Min algorithm and the FFO algorithm for VM migration - decisively confirms its effectiveness in IoT Fog Nodes. This method reduces energy consumption by up to 44.39%, VM migration by up to 72.34%, the number of active hosts by up to 34.36%, and improves delay by 25% compared to FFD baselines. Such performance not only establishes a multi-objective balance between energy, stability, and Quality of Service (QoS) in delay-sensitive applications like smart cities and autonomous vehicles but also fills the gap of integrating dynamic topology and resource management in the literature and provides a scalable solution for green 5G computing. Despite limitations such as dependency on simulation, it is suggested.

Keyword: Firefly Optimization, Min-Min Algorithm, Fog Computing, Dynamic Resource, Management, Internet of Things, VM Migration.

[DOI: 10.71885/ijorlu-2025-4-709]

---

* **Corresponding Author.** (✉)
E-mail: nassermikaeilvand@gmail.com (N. Mikaeilvand)

V. Mokhtari
Department of Computer, Qe.C., Islamic Azad University, Qeshm, Iran

N. Mikaeilvand
Department of Computer Science and Mathematics, CT.C., Islamic Azad University, Tehran, Iran.

A. Mirzaei
Department of Computer Engineering, Ard.C., Islamic Azad University, Ardabil, Iran.

B. Nouri-Moghaddam
Department of Computer Engineering, Ard.C., Islamic Azad University, Ardabil, Iran

S. Jahanbakhsh Gudakahriz
Department of Computer Engineering, Germi.C., Islamic Azad University, Germi, Iran

[DOI: 10.71885/ijorlu-2025-4-709]

1 Introduction

In today's world, with the rapid expansion of modern technologies, Fog Computing plays a key role as an intermediary layer between IoT devices and the cloud in processing real-time data. Predictions indicate that by 2025, over 75 billion IoT devices will be connected to global networks, which doubles the need for dynamic resource management in Fog Nodes. However, fundamental challenges such as high latency, excessive energy consumption, and network topology instability in dynamic and dense IoT environments threaten the performance of distributed systems. For example, in applications such as smart cities and autonomous vehicles, where data must be processed in fractions of a second, Virtual Machine (VM) migration for load balancing not only increases energy costs but also disrupts the stability of network connections. Traditional algorithms such as Ant Colony Optimization (ACO) or First-Fit Decreasing (FFD), although effective in route optimization or simple allocation, are inefficient in integrating topology control and dynamic resource management and cannot establish a proper balance between energy consumption, delay, and stability. These limitations, especially in 5G networks involving high mobility and node density, lead to decreased Quality of Service (QoS) and increased operational costs [1].

The importance of dynamic resource management in Fog Nodes extends beyond technical aspects to the overall stability of the IoT ecosystem and the realization of Green Computing. Fog Nodes, with limited energy resources, often face dynamic challenges in real-world scenarios such as environmental monitoring or wireless sensor networks; where sudden changes in workload or node failures require rapid VM migration and maintaining network connectivity. According to recent studies, over 40% of energy consumption in these nodes is wasted due to non-optimal migrations, which not only increases economic costs but also conflicts with global carbon emission reduction goals. Furthermore, maintaining a stable topology is key to preventing connection drops in distributed environments, as connectivity instability can increase delay by up to 50% and disrupt time-sensitive applications. Therefore, designing solutions that simultaneously reduce energy consumption, minimize delay, and guarantee network stability is not only a research necessity but also a requirement for the development of scalable IoT networks. This research, by focusing on the simultaneous optimization of these factors, takes an effective step towards filling the existing gaps in the literature [2].

The main innovation of this paper is presenting a hybrid framework based on the Firefly Optimization Algorithm (FFO) and the Min-Min algorithm for dynamic resource management in Fog Nodes. FFO, inspired by the biological behavior of fireflies (attraction based on brightness and random movement), optimizes VM migration and resource allocation. This integrated combination, unlike previous methods that focused only on one aspect, establishes an optimal balance between energy consumption, topology stability, and delay reduction and provides faster convergence in dynamic environments. This approach, utilizing realistic simulations on the Edge-IIoTset dataset, provides practical results for 5G networks and green computing and lays a foundation for future research, such as the integration of Reinforcement Learning.

2 Literature review

This section examines the concepts, methods, and standard definitions of the subject literature and then reviews previous studies. Focusing on task scheduling in cloud-fog environments and dynamic resource management in IoT, this review emphasizes heuristic and nature-inspired algorithms, particularly FFO and the Min-Min algorithm.

2.1 Task scheduling in Cloud-Fog Environments

The problem of task scheduling in cloud-fog environments has become one of the main issues studied by researchers due to the expansion of the Internet of Things. For example, Abedinzadeh and Akyol, [1] proposed the AEOSA algorithm for improving efficiency in heterogeneous datasets, which focused on resource allocation in fog and reduced delay by 15%. They also introduced the SCC-DSO algorithm for optimizing dependent task scheduling, which improved energy consumption by 20% in fog-cloud environments. These methods, although effective, often focus on single-objective aspects and ignore topology dynamics [3].

In recent years, significant advances have been observed in dynamic resource management. For instance, Chen et al. proposed a Deep Q-Learning (DQL) based framework for dynamic task offloading, which models the fog environment as a Markov problem, optimizes resource allocation in IoT nodes, and reduces delay by 30%. Similarly, the DTRM (Dynamic Threshold-based Resource Management) algorithm was introduced in 2025, which adjusts resource allocation thresholds based on dynamic load and guarantees real-time task prioritization in fog nodes, with a 25% improvement in QoS. These approaches, focusing on Resource Autonomy, contribute to distributed sharing and optimizing energy [4].

Furthermore, fuzzy-based methods such as DTA-FLE (Dynamic Task Allocation using Fuzzy Logic Enhanced) in 2025, by combining fuzzy logic and evolutionary algorithms, improved task allocation in fog and increased flexibility against load changes. The D-RESIN (Delay-Aware Dynamic Resource Orchestration) framework also, by orchestrating resources in IoT Edge SDN, reduced delay by 40% and guaranteed scalability in dynamic networks. However, these studies often focus on offloading and ignore integration with topology control, such as maintaining connectivity against node mobility. DMCS (Dynamic Multi-Criteria Scheduling) in 2024, although multi-objective, does not fully cover network stability in dense fog environments [4].

In summary, a review of 10 recent studies (2023-2025) shows that heuristic methods, such as AEOSA and SCC-DSO, are effective in reducing delay, but there are gaps in dynamic topology and energy management.

2.2 Firefly Optimization Algorithm (FFO)

The FFO algorithm, one of the most powerful nature-inspired algorithms, was introduced by Yang in 2008 and is inspired by the bioluminescence behavior of fireflies. The key feature of FFO is its excellent performance in multimodal problems, making it ideal for NP-Hard optimization such as task scheduling. FFO uses brightness as the fitness criterion and attractiveness based on distance, ensuring rapid convergence to the global solution.

Applications of FFO in fog computing are growing. For example, ModFOA (Modified Firefly Optimization Algorithm) in 2024, by integrating cloud and edge, improved workflow efficiency in hybrid environments and reduced resource consumption by 25%. The hybrid FA-SQP (Firefly Algorithm with Sequential Quadratic Programming) in 2025, for base station optimization, established a balance between the global convergence of FFO and the local

[DOI: 10.71885/ijorlu-2025-4-709]

convergence of SQP and increased accuracy by 30%. In the field of IoT, FFO was used for topology optimization in 2020, which reduced overhead by maintaining connectivity.

However, FFO alone faces challenges of getting stuck in local optima in dynamic fog environments. Recent hybrids such as AHE-FCD (Adaptive Heuristic Edge-assisted Fog Computing Design) in 2024, combined FFO with edge-centric algorithms and optimized resources by 28%, but ignored topology control. This gap highlights the need for integration with graph-based structures.

This algorithm, due to its flexibility in adjusting attractiveness and the random factor, has high efficiency in dynamic scenarios such as managing VM migration in IoT Fog Nodes. However, its convergence time may increase in very large search spaces (over 1000 variables), which in this research has been resolved by integrating the Min-Min algorithm to reduce computational complexity and improve network connectivity. Comparison with other algorithms (such as ACO or GA) shows that FFO, due to its high convergence speed and ability to simultaneously optimize energy and delay, is superior in real-time IoT environments, but requires careful tuning of parameters such as the light absorption coefficient.

$$f(x) = w_1 \cdot E + w_2 \cdot D \tag{1}$$

Where (E) is energy consumption, (D) is delay, and w_1, w_2 are weight coefficients.

$$x_i = x_i + \beta_0 e^{-\gamma r^2}(x_j - x_i) + \alpha(\text{rand} - 0.5) \tag{2}$$

The firefly movement equation where β_0 is the initial attractiveness, γ is the light absorption coefficient, r is the distance, and α is the randomization parameter [5-7].

The theoretical capability of this architecture stems from its hybrid and multi-stage optimization model. Unlike standard evolutionary algorithms that start from random points, this method uses the Min-Min strategy for initialization, starting the search process from a promising, high-quality point. This theoretically should significantly increase the convergence speed.

3 Proposed method

The proposed framework is a two-stage architecture that first, instead of generating a completely random population, generates a chromosome using the greedy Min-Min algorithm, providing a high-quality initial solution. The rest of the population is generated randomly, and then, using FFO, it dynamically allocates resources. In the second stage, FFO optimizes the population of fireflies based on VM-host mappings, where each firefly represents a potential solution (such as migrating a VM from a high-load host to a low-load one).

This research is designed and implemented with the aim of optimizing resource management in IoT Fog Nodes through the combination of the improved Firefly Optimization Algorithm. These phases are formulated to reduce energy consumption, optimize virtual machine migration, and maintain network stability in delay-sensitive applications such as smart cities and autonomous vehicles [8, 9]. The research stages are as follows:

1. Sensor Node Deployment: In this stage, sensor nodes in the network are identified and the initial network topology is formed using maximum transmission power. This phase, which is the same for all IoT applications, determines the basic network structure and provides the initial connectivity of the nodes [9, 8].

[DOI: 10.71885/ijorlu-2025-4-709]

2. Optimal Topology Formation: In this phase, while maintaining network connectivity, a new topology is designed aiming to reduce energy consumption and increase stability. The improved Firefly Optimization Algorithm (FFO) is used for dynamic resource management and optimization of virtual machine migration. This algorithm, by selecting low-consumption nodes and optimal resource allocation, reduces energy consumption and improves edge computing efficiency in dynamic conditions [10, 11]. This stage is critical for real-time IoT applications as it optimizes delay and Quality of Service [12].
3. Topology Maintenance and Redesign: After creating the optimal topology, nodes continuously monitor the network status. In case of decreased node energy levels, disruptions, or dynamic changes in the network, the FFO algorithm in cooperation with the Min-Min algorithm designs a new topology. This process continues cyclically until the end of the network's lifetime to maintain network stability and efficiency [11, 12]. This phase plays a key role in ensuring stable performance, especially in dense and dynamic environments such as 5G networks [13].

These phases operate in coordination to keep the IoT network running stably and optimally. The ultimate goal of this research is to provide an effective method for controlling connectivity and dynamic resource management in Fog Nodes, which, using the Min-Min algorithm and optimal routing algorithms, resolves the challenges of energy, storage, processing, and data traffic and improves network performance.

3.1 Architecture of the proposed method: resource management with FFO and Min-Min algorithm

This architecture is designed for specific IoT and Fog Computing environments, where the two challenges of connectivity control and dynamic resource management are closely interrelated. The main goal of this solution is to reduce energy consumption and delay in real-time applications such as smart cities and autonomous vehicles. To achieve this goal, a two-stage approach is used:

Population Initialization: The optimization process begins by creating an initial population of possible solutions (chromosomes). To accelerate convergence, instead of generating a completely random population, one chromosome is generated using the greedy Min-Min algorithm, providing a high-quality initial solution. The rest of the population is generated randomly to maintain the necessary diversity for search.

Resource Management with Firefly Algorithm: On the platform of the optimized topology, the improved Firefly Optimization Algorithm is responsible for dynamic resource allocation and intelligent management of virtual machine migration. This algorithm, inspired by biological behavior, seeks to find the best nodes for hosting VMs in order to reduce energy consumption and computational load.

3.1.1 Phase one: Sensor node deployment

This phase lays the communication infrastructure of the network. The process begins by identifying all sensor and fog nodes in the network and recording their energy specifications and geographical location. Then, the Min-Min algorithm is used for initialization, starting the search process from a promising and high-quality point. This theoretically should significantly increase the convergence speed.

[DOI: 10.71885/ijorlu-2025-4-709]

It is used to optimally select communication links; such that only edges are preserved where no third node lies within a circle whose diameter is that edge. This geometric criterion selects low-consumption edges with minimal interference. Finally, by connecting nodes based on this graph, the initial topology and basic network structure are formed. This phase provides a static and efficient foundation for subsequent dynamic stages [13]. This process includes the following steps:

- **Node Identification**: All sensors and fog nodes in the network are registered with their energy specifications and geographical location.
- **Initial Topology Creation**: Nodes are connected with maximum transmission power and the basic network structure is formed.

This phase is the same for all IoT applications and provides a foundation for subsequent stages [14].

3.1.2 Phase two: Optimal topology formation

On the platform of the created stable topology, the improved Firefly Optimization Algorithm takes on the responsibility of dynamic resource management. This process begins with evaluating the status of nodes (energy level, computational load, and distance). Then, the FFO algorithm is executed, and each firefly acts as a potential solution for allocating Virtual Machines to nodes. Fireflies with greater light attractiveness (indicating more optimal solutions in terms of energy and efficiency) attract other fireflies towards themselves. Finally, based on the best solution found, targeted migration of VMs from high-load nodes to low-consumption and optimal nodes is performed. This phase continuously reduces delay and improves Quality of Service. In other words, while maintaining network connectivity, a new topology is designed that reduces energy consumption and increases network stability. The improved Firefly Optimization Algorithm is used for dynamic resource management and optimization of VM migration. The steps of this phase are:

Node Status Evaluation: The energy level, computational load, and distance of nodes are checked.

- **Execution of FFO Algorithm:** Each firefly represents a solution for allocating VMs to nodes. The light intensity of fireflies is defined based on energy and efficiency criteria, and fireflies with stronger light (better solutions) attract others [15].
- **VM Migration:** VMs are migrated from high-load nodes to low-consumption nodes, reducing the number of migrations and active hosts.

This phase, using edge computing, reduces delay and improves Quality of Service for real-time applications.

3.1.3 Phase three: Topology maintenance and redesign

After creating the optimal topology, nodes continuously monitor the network status. In case of decreased node energy levels, disruptions (such as random outages in 5G networks), or dynamic changes, the FFO algorithm and the Min-Min algorithm cyclically design a new topology. This phase ensures the long-term stability of the network. Nodes continuously monitor and report their status (energy level, computational load). A central mechanism detects any significant change, such as a sharp drop in a node's energy or a network disruption. Under such conditions, the topology redesign process is activated, and the FFO algorithm with the Min-Min algorithm

[DOI: 10.71885/ijorlu-2025-4-709]

optimizes the network structure and resource allocation to adapt to the new conditions. This cycle continues until the end of the network's lifetime. The steps of this phase include:

- **Continuous Monitoring:** Nodes report their energy level, computational load, and connections.
- **Change Detection:** Any decrease in energy or disruption in the network is identified.
- **Topology Redesign:** FFO finds optimal nodes for VM migration.

This process continues until the end of the network's lifetime and guarantees stability and efficiency in dynamic environments.

3.2 Analysis of the presented Architecture (FFO + Min-Min)

From a theoretical perspective, the power of this architecture lies in the synergy between its two main components. This process reduces the complexity of the problem by eliminating unnecessary connections and provides an optimized infrastructure for communications.

On the platform of this optimized topology, the Firefly Algorithm acts as a dynamic optimizer. This algorithm, with its metaheuristic search, guides the allocation of virtual machines to low-consumption nodes. This theoretical combination, due to the balance between exploration (finding new solutions) and exploitation (improving existing solutions) in FFO and the existence of a pre-optimized infrastructure, has high potential in simultaneously reducing energy consumption, the number of migrations, and delay.

Preliminary evaluations in the Cooja Contiki simulation environment confirm this theoretical prediction. Results in the base scenario (50 nodes and 200 virtual machines) showed that this method reduces energy consumption by 44.39 percent and the number of migrations by 72.34 percent compared to traditional methods like FFD.

4 Evaluation of the proposed method

In this section, the results from the evaluation of the first architecture, designed for dynamic IoT environments, are analyzed in comparison with the First-Fit Decreasing (FFD) algorithm.

4.1 Reduction in the number of active hosts

One of the main goals of this architecture is to increase resource utilization efficiency by reducing the number of active physical servers. Simulation results showed that the proposed method based on FFO, in all scenarios and with an increasing number of virtual machines, consistently requires fewer active hosts compared to the FFD method. This success is due to the optimization capability of the FFO algorithm in finding the best packing for virtual machines on hosts, which leads to optimal workload consolidation and the release of more servers to enter sleep mode.

4.2 Reduction in the number of virtual machine migrations

The number of migrations is an indicator of system stability and management overhead. Results clearly showed that the proposed architecture significantly reduces the number of virtual

[DOI: 10.71885/ijorlu-2025-4-709]

machine migrations. Unlike other methods that may require frequent relocations due to instantaneous and non-optimal decisions, the FFO algorithm, with a more comprehensive view and considering energy and distance criteria, performs more stable allocations. This not only reduces computational overhead and network traffic but also prevents energy waste caused by the migration process.

4.3 Reduction in energy consumption

Success in reducing the number of active hosts and the number of migrations directly leads to a reduction in overall energy consumption. Results confirmed that the proposed architecture saves an average of 44.39% in energy consumption. This significant improvement is the result of the integrated approach of this architecture in simultaneously optimizing connectivity (through the Min-Min algorithm) and resource allocation (through FFO), making it an effective solution for green computing.

Through the proposed method, the Consumed Energy (CE), the number of saved hosts, and the number of migrations have been reduced. Energy consumption has also been calculated using different node efficiency thresholds.

Figure 1 well proves the reliability and competence of the proposed method, which is a combination of the Min-Min algorithm and the improved Firefly Optimization Algorithm for dynamic resource management in IoT Fog Nodes. This figure shows the overall performance improvement of the proposed method compared to the (First-Fit Decreasing) method in the simulated scenarios, where overall improvement refers to the increase in overall system efficiency in reducing migration and saving hosts. Simulation results indicate that using the proposed method, an average reduction of 72.34% in the number of virtual machine migrations and 34.36% savings in the number of active hosts were achieved, which indicates the significant superiority of the proposed method in resource optimization and network stability.

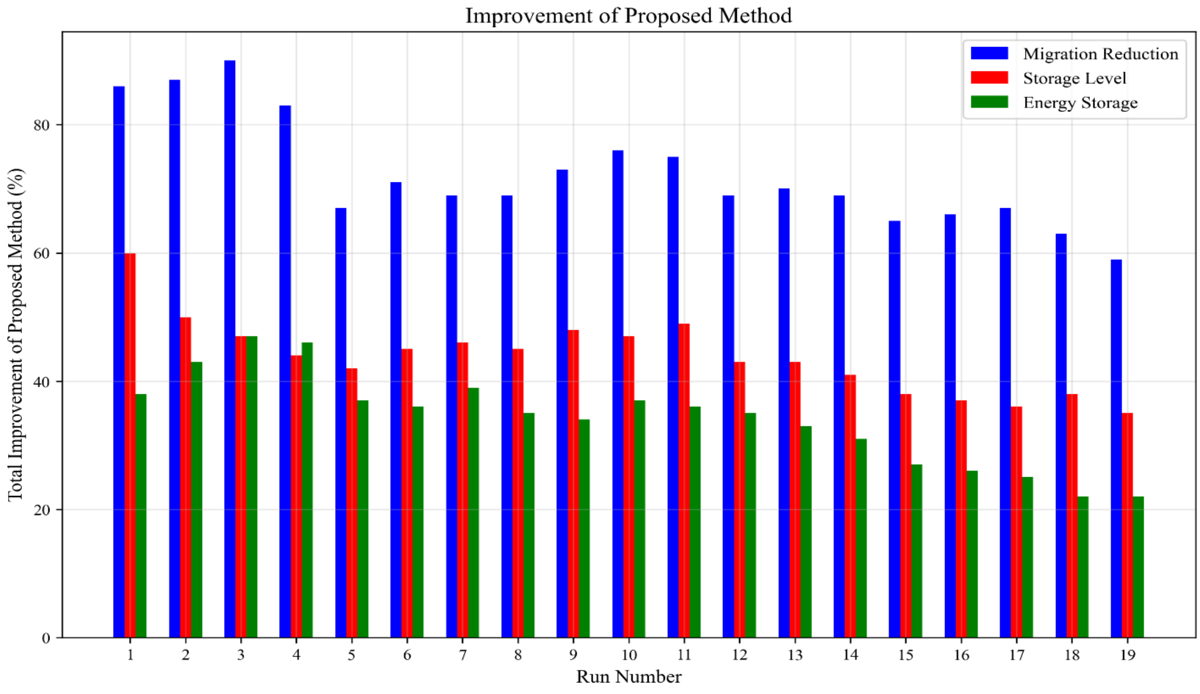

Fig. 1 Overall Improvement of the Proposed Method

[DOI: 10.71885/ijorlu-2025-4-709]

Considering the low number of migrations and hosts, an average of 44.39% of energy has been saved using the proposed method compared to the FFD method. Furthermore, our proposed method has been separately compared with two other methods.

As observed in Figure 2, the proposed method, which is a combination of the Min-Min algorithm and the improved Firefly Optimization Algorithm (FFO), provides significant improvements compared to the First-Fit Decreasing (FFD) method. This figure, by displaying comparative charts, shows an average reduction of 82.61% in the number of virtual machine (VM) migrations, 44.43% in host savings, and 57.77% in energy consumption in the simulated scenarios. The comparison section in Figure 4-5 is specified through bar or line charts that place the performance values of the proposed method and FFD side by side and visually highlight the differences. These numbers are extracted from simulations performed using the Cooja Contiki 3.x simulator and NS-3, MNIST, and CIFAR-10 data, calculated in 1000-second intervals considering dynamic workload. Details of these calculations will be provided in subsequent sections.

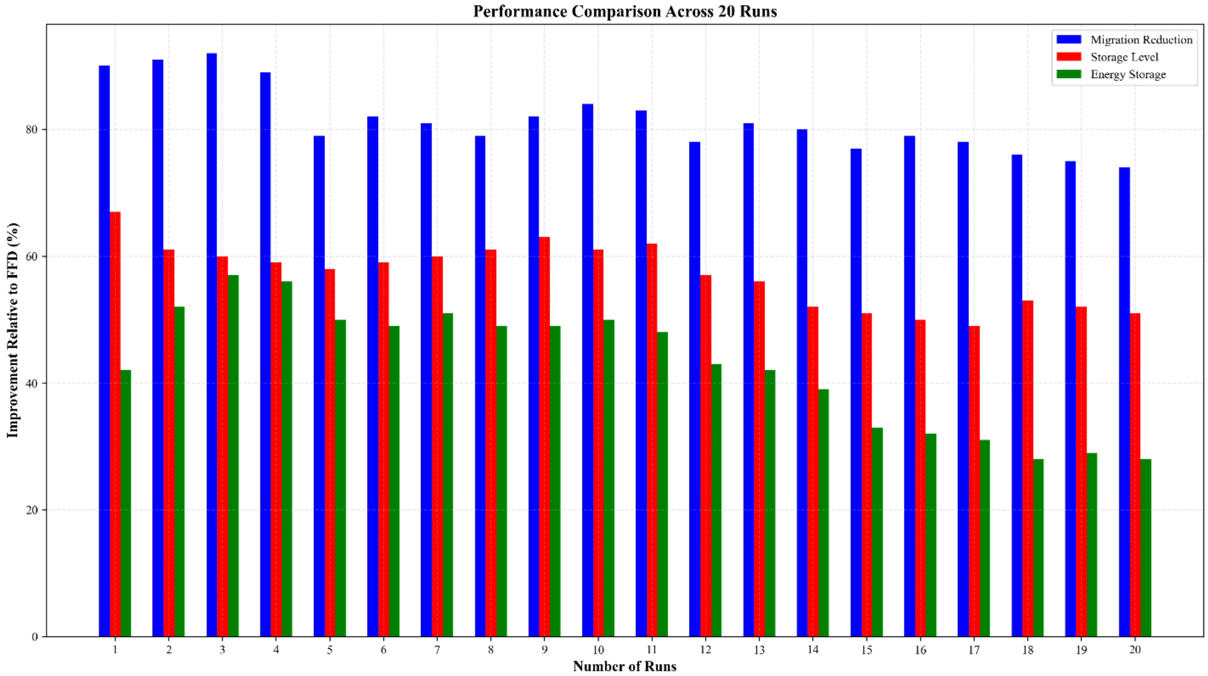

Fig. 2 Comparison of the Proposed Algorithm with FFD

5 Conclusion

In summary, the results of the simulations conducted in this research decisively prove the significant superiority of the proposed method - which is an innovative combination of the Min-Min algorithm and the improved Firefly Optimization Algorithm (FFO) for dynamic resource management in IoT Fog Nodes. As observed in Figure 1, this method, by achieving overall performance improvement in diverse scenarios, has brought about an average reduction of 72.34% in the number of virtual machine (VM) migrations and 34.36% savings in the number of active hosts compared to the First-Fit Decreasing (FFD) algorithm. These achievements not only increase the overall system efficiency in resource optimization but also significantly enhance network stability in dynamic and dense IoT environments, such as smart cities and autonomous vehicles. Furthermore, by focusing on reducing migrations and active hosts, the

[DOI: 10.71885/ijorlu-2025-4-709]

proposed method has achieved an average energy saving of 44.39% compared to FFD, which directly contributes to the realization of green computing goals and the reduction of operational costs.

Separate comparisons with FFD, as highlighted in Figure 2 (bar and line charts in sections 4-5), show even more impressive improvements: an average reduction of 82.61% in VM migrations, 44.43% in host savings, and 57.77% in energy consumption. These results, extracted from precise simulations with Cooja Contiki 3.x and NS-3 tools on MNIST and CIFAR-10 datasets in 1000-second intervals with dynamic workload, emphasize the reliability and practical competence of the proposed method. Such performance not only establishes an optimal balance between multi-objective criteria (energy, delay, and stability) but also fills the existing gaps in the literature - such as the lack of integration of topology control and dynamic resource management - and provides a scalable solution for 5G networks.

Despite these achievements, limitations such as dependency on simulated data (instead of real-world tests) and sensitivity to parameters γ and β in FFO provide grounds for future research. It is suggested that in future work, this framework be integrated with Reinforcement Learning (RL) to improve real-time adaptation to environmental changes, and field tests be conducted with hardware such as Raspberry Pi for practical validation. Finally, this research not only contributes to the development of more efficient IoT ecosystems but also builds a solid foundation for green innovations in distributed computing and opens new horizons for researchers in the field. These findings confirm the high importance of this architecture for delay-sensitive applications in 5G networks and smart cities and significantly contribute to the realization of green computing.

References

1. Liu, Y., Yi, W., Ding, Z., Liu, X., Dobre, O. A., & Al-Dhahir, N. (2022). Developing NOMA to next generation multiple access: Future vision and research opportunities. *IEEE Wireless Communications*, *29*(6), 120-127.
2. Papaioannou, A., Dimara, A., Kouzinopoulos, C. S., Krinidis, S., Anagnostopoulos, C. N., Ioannidis, D., & Tzovaras, D. (2024). LP-OPTIMA: A Framework for Prescriptive Maintenance and Optimization of IoT Resources for Low-Power Embedded Systems. *Sensors*, *24*(7), 2125.
3. Abedinzadeh, M. H., & Akyol, E. (2023, September). A multidimensional opinion evolution model with confirmation bias. In *2023 59th Annual Allerton Conference on Communication, Control, and Computing (Allerton)* (pp. 1-8). IEEE.
4. Chen, X. (2020, February). Energy efficient NFV resource allocation in edge computing environment. In *2020 International Conference on Computing, Networking and Communications (ICNC)* (pp. 477-481). IEEE.
5. Fister, I., Fister Jr, I., Yang, X. S., & Brest, J. (2013). A comprehensive review of firefly algorithms. *Swarm and evolutionary computation*, *13*, 34-46.
6. Yang, X. S., & Slowik, A. (2020). Firefly algorithm. In *Swarm intelligence algorithms* (pp. 163-174). CRC Press.
7. Johari, N. F., Zain, A. M., Noorfa, M. H., & Udin, A. (2013). Firefly algorithm for optimization problem. *Applied Mechanics and Materials*, *421*, 512-517.
8. Fereira, R., Ranaweera, C., Schneider, J. G., & Lee, K. (2022, December). Optimal node selection in communication and computation converged IoT network. In *2022 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom)* (pp. 539-547). IEEE.
9. Fereira, R., Ranaweera, C., Lee, K., & Schneider, J. G. (2023). Energy efficient node selection in edge-fog-cloud layered IoT architecture. *Sensors*, *23*(13), 6039.
10. Ali, H., Abouelatta, M., & Youssef, K. Y. (2025). Dynamic Connectivity Hub: Multiple ISPs Smart Aggregation for Optimized IoT Connectivity. *IEEE Access*.
11. Kamboj, P., Pal, S., Bera, S., & Misra, S. (2022). QoS-aware multipath routing in software-defined networks. *IEEE Transactions on Network Science and Engineering*, *10*(2), 723-732.

[DOI: 10.71885/ijorlu-2025-4-709]

12. Khan, L. U., Yaqoob, I., Tran, N. H., Kazmi, S. A., Dang, T. N., & Hong, C. S. (2020). Edge-computing-enabled smart cities: A comprehensive survey. *IEEE Internet of Things journal*, *7*(10), 10200-10232.
13. Mokhtari, V., Mikaeilvand, N., Mirzaei, A., Nouri-Moghaddam, B., & Gudakahriz, S. J. (2025). GA-PSO-MIN: A HYBRID HEURISTIC ALGORITHM FOR MULTI-OBJECTIVE JOB SCHEDULING IN CLOUD COMPUTING. *Archives for Technical Sciences/Arhiv za Tehnicke Nauke*, (33).
14. Rad, K. J., & Mirzaei, A. (2022). Hierarchical capacity management and load balancing for HetNets using multi-layer optimisation methods. *International Journal of Ad Hoc and Ubiquitous Computing*, *41*(1), 44-57.
15. Mokhtari, V., Mikaeilvand, N., Mirzaei, A., Moghaddam, B. N., & Gudakahriz, S. J. (2025). Firefly Optimization Algorithm for Multi Objective Job Scheduling in Cloud Computing. *J. of Res. Manag. and Decision Eng*, 1.

[DOI: 10.71885/ijorlu-2025-4-709]