International Journal of Advance Scientific Research
(ISSN – 2750-1396)
VOLUME 06 ISSUE 03 Pages: 79-88
OCLC – 1368736135

Journal Website: http://sciencebring.com/index.php/ijasr

Research Article

A Comprehensive Framework for Many-Objective Resource Scheduling and Carbon-Aware Orchestration in Distributed Fog-Cloud Ecosystems

Submission Date: February 01, 2026, **Accepted Date:** February 28, 2026,
Published Date: March 31, 2026

James Trachen
Department of Computer Science and Information Systems, University of Melbourne, Australia

Abstract

The rapid proliferation of Internet of Things (IoT) devices and the subsequent data deluge have necessitated a paradigm shift from centralized cloud computing to distributed architectures encompassing fog and edge computing. As these environments evolve, the complexity of task scheduling has transitioned from simple multi-objective optimization to many-objective optimization, where conflicting goals-such as energy efficiency, reliability, deadline constraints, cost, and carbon footprint-must be balanced simultaneously. This research article provides an extensive investigation into the theoretical underpinnings and practical applications of many-objective evolutionary algorithms and swarm intelligence within the context of fog-integrated cloud environments. By synthesizing contemporary research on diversity assessment, grid-based evolutionary strategies, and carbon-aware Kubernetes-native scheduling, this study proposes a holistic framework for managing scientific workflows and big data pipelines. The analysis explores the nuances of balanceable fitness estimation, clustering-based selection mechanisms, and the integration of data analytics for congestion management. Special emphasis is placed on the emerging necessity of carbon-aware scheduling to mitigate the environmental impact of large-scale distributed systems. The findings suggest that while traditional meta-heuristics are effective for low-dimensional problems, the scalability of many-objective approaches, such as those implemented in the PlatEMO platform, is essential for the next generation of smart grid and industrial IoT applications.

Keywords

Fog Computing, Many-Objective Optimization, Cloud Task Scheduling, Carbon-Aware Computing, Evolutionary Algorithms, IoT Resource Allocation.

INTRODUCTION

The contemporary digital landscape is defined by an unprecedented convergence of decentralized computing resources and high-velocity data streams. As the Internet of Things (IoT) matures, the traditional reliance on centralized cloud data centers has revealed significant bottlenecks, particularly regarding latency, bandwidth consumption, and energy overhead. To address these challenges, the concept of fog computing has emerged as a critical intermediary layer, extending cloud capabilities closer to the data source (Aazam et al., 2018). However, the introduction of this layer significantly complicates the problem of resource allocation and task scheduling. In these distributed environments, scheduling is no longer a matter of simply minimizing execution time; it involves a multifaceted trade-off between conflicting objectives that often exceed the capacity of standard multi-objective optimization techniques.

The shift from multi-objective to many-objective optimization-where the number of objectives typically exceeds four-introduces the "curse of dimensionality." Traditional Pareto-based approaches, which rely on dominance relationships, lose their selective pressure as most solutions become non-dominated in high-dimensional spaces. This necessitates advanced diversity assessment methods to ensure that the evolutionary process explores the objective space effectively without converging prematurely (Wang, Jin, & Yao, 2017). Furthermore, the integration of fog and cloud environments requires mobility-aware application scheduling, as the physical movement of IoT devices can disrupt service continuity (Bittencourt et al., 2017).

Beyond technical performance metrics, the global imperative for environmental sustainability has introduced carbon awareness as a primary objective in system design. Recent frameworks, such as Carbon-Kube, have begun to address the need for Kubernetes-native scheduling that prioritizes the reduction of carbon intensity in big data pipelines (Bhat, Sirikonda, Katoch, & Jain, 2026). This research article explores the integration of these diverse requirements-ranging from the mechanical reliability of smart factory applications (Dehnavi et al., 2019) to the economic optimization of virtual machine migration (Ruan et al., 2019). By reviewing the evolution of grid-based and clustering-based evolutionary algorithms, this study aims to establish a theoretical foundation for more resilient and sustainable distributed computing ecosystems.

Theoretical Foundations of Many-Objective Optimization The foundation of modern task scheduling in distributed systems lies in the mathematical formulation of many-objective optimization problems (MaOPs). Unlike traditional optimization, MaOPs deal with a large number of conflicting criteria, which makes the visualization and selection of optimal solutions increasingly difficult. One of the primary hurdles identified in the literature is the loss of selection pressure in Pareto-based algorithms. As the number of objectives increases, the proportion of the population that is non-dominated grows exponentially, leading to a breakdown in the ability

International Journal of Advance Scientific Research
(ISSN – 2750-1396)
VOLUME 06 ISSUE 03 Pages: 79-88
OCLC – 1368736135

of the algorithm to distinguish between "good" and "better" solutions.

To combat this, researchers have proposed grid-based evolutionary algorithms (GrEA). By partitioning the objective space into a grid, these algorithms can maintain a high level of diversity among solutions by using grid-based fitness values rather than simple Pareto dominance (Yang et al., 2013). This approach allows for a more granular assessment of how solutions are distributed across the potential performance frontier. Similarly, the development of the PlatEMO platform has provided researchers with a standardized environment to test and compare these complex evolutionary multi-objective optimization strategies (Tian, Cheng, Zhang, & Jin, 2017).

Another significant advancement is the introduction of balanceable fitness estimation. In particle swarm optimization (PSO), which is frequently used for resource allocation due to its computational efficiency, balancing convergence and diversity is a perpetual challenge. Lin et al. (2018) demonstrated that by incorporating a balanceable fitness estimation mechanism, PSO could be adapted for many-objective environments, ensuring that the swarm does not collapse into a localized region of the search space. This is particularly relevant for scientific workflow scheduling in the cloud, where the search space is vast and highly discontinuous (Saeedi et al., 2020).

Furthermore, the role of clustering-based evolutionary algorithms cannot be overstated. By grouping similar solutions together, these algorithms can select representative candidates from each cluster, thereby preserving the structural diversity of the population (Lin et al., 2019). This is especially useful in cloud computing environments where task dependencies-such as those found in complex scientific workflows-require the optimizer to account for both the individual performance of a task and its impact on the overall execution chain (Alkhanak et al., 2018).

Resource Allocation in Fog and Cloud Ecosystems
The integration of fog computing into the cloud hierarchy introduces a heterogeneous resource pool that varies significantly in terms of processing power, storage capacity, and network stability. Effective resource allocation in this "Cloud of Things" requires algorithms that can handle the volatility of IoT data (Tortonesi et al., 2019). For instance, in healthcare applications, energy-aware fog-enabled systems must ensure that life-critical data is processed with high reliability and low latency, even if it means incurring higher economic costs (Mahmoud et al., 2018).

Scientific workflow scheduling represents one of the most demanding tasks in this domain. Unlike independent task scheduling, workflows involve complex Directed Acyclic Graphs (DAGs) where the output of one task serves as the input for another. Optimization here involves minimizing both the "makespan" (total execution time) and the total cost of renting cloud resources (Zhou et al., 2019). Recent developments have seen the application of hyper-heuristic cost optimization approaches, which aim to find the best heuristic for a given workflow structure rather than applying a one-size-fits-all algorithm (Alkhanak et al., 2018).

In the context of the smart factory, the stakes are even higher. Industrial applications require reliability-aware resource provisioning to prevent system downtime that could result in significant physical or financial damage (Dehnavi et al., 2019). This necessitates a move toward real-time scheduling where the algorithm can adapt to the sudden failure of a fog node or a sudden spike in data traffic. Data analytics-based particle swarm optimization has shown promise in this area, particularly for determining congestion thresholds in low-voltage networks, which is a critical component of smart grid management (Leiva, Pardo, & Aguado, 2019).

The allocation of virtual machines (VMs) also plays a central role in cloud efficiency. Algorithms inspired by nature, such as ant colony optimization, have been adapted to optimize VM placement based on multiple objectives, including hardware utilization and energy consumption (Yuping, 2019). Ruan et al. (2019) further refined this by focusing on the performance-to-power ratio, suggesting that migration strategies should prioritize nodes that offer the best computational output per unit of energy consumed. This transition toward "green" cloud computing is a recurring theme in the literature, reflecting a broader societal shift toward environmental responsibility.

Carbon-Awareness and Sustainability in Distributed Systems As the energy consumption of data centers continues to rise, the academic community has begun to focus on the environmental footprint of computing. The concept of carbon-aware scheduling goes beyond simple energy efficiency. It involves scheduling tasks during times when the local power grid is supplied by renewable energy sources or moving workloads to geographic regions where the carbon intensity of electricity is lower.

The Carbon-Kube framework represents a significant step in this direction. By integrating carbon-intensity data directly into the Kubernetes scheduler, big data pipelines can be orchestrated in a way that minimizes their carbon footprint without significantly compromising on performance (Bhat et al., 2026). This requires a sophisticated multi-objective approach that treats "grams of CO2 emitted" as a primary metric alongside latency and cost.

This sustainability focus is also evident in the management of smart grids. Multi-objective load scheduling in smart grids aims to balance the demand of consumers with the available supply from both traditional and renewable sources (Sadhukhan & Sivasubramani, 2018). Electric vehicle (EV) integration adds another layer of complexity; the scheduling of EV charging must be managed to avoid distribution system overloads while also satisfying the owners' requirements for battery state-of-charge (Singh & Tiwari, 2018). These problems are inherently many-objective, as they involve thousands of individual nodes, each with its own constraints and preferences.

Task Scheduling and Optimization Strategies In cloud environments, the Deadline and Cost Optimization based Task Scheduling (DCOTS) approach exemplifies the tension between user requirements and provider profitability (Grewal & Mangla, 2023). Users typically demand that their tasks be completed within a specific timeframe (deadline), while also wishing to minimize their expenditure. Providers, on the other hand, want to maximize resource utilization and minimize operational costs. Evolutionary multi-factor algorithms have been proposed to solve these multi-objective cloud task scheduling problems, allowing the system to handle multiple tasks across different domains simultaneously (Cui et al., 2023).

Energy awareness is not just an environmental concern; it is also a cost and reliability concern. In systems that utilize Dynamic Voltage and Frequency Scaling (DVFS), the scheduler can adjust the clock speed of processors to save energy during periods of low demand (Hassan et al., 2020). However, reducing the frequency of a processor can increase the execution time of a task, potentially leading to deadline violations. A smart energy and reliability-aware scheduling algorithm must therefore find the "sweet spot" where energy is saved without compromising the stability of the workflow.

The use of big data analytics in smart homes further illustrates the need for efficient fog-cloud collaboration. By processing data locally at the fog level, smart home systems can provide real-time responses to residents while sending only the necessary summarized data to the cloud for long-term storage and trend analysis (Yassine et al., 2019). This hierarchical approach reduces the "data deluge" and ensures that the communication network remains functional even as the number of smart devices grows (Tortonesi et al., 2019).

Methodology

The methodology for addressing many-objective optimization in this research centers on the synthesis of evolutionary computation and swarm intelligence. The process begins with the definition of the objective space. In a typical fog-cloud scheduling scenario, we consider a set of tasks T={t1,t2,...,tn} and a set of heterogeneous resources R={r1,r2,...,rm}. The primary objectives include:

1. Makespan Minimization: The total time required to complete all tasks.

2. Cost Reduction: The total monetary cost incurred for resource usage.

3. Energy Consumption: The total kilowatt-hours used by the physical hardware.

4. Reliability Maximization: The probability that all tasks will complete without failure.

5. Carbon Footprint: The total carbon emissions associated with the energy source.

6. Load Balancing: The degree of uniformity in resource utilization across the network.

To solve this, we employ a clustering-based evolutionary algorithm (CBEA). The core logic of CBEA involves using a K-means or fuzzy C-means clustering technique to partition the population of potential schedules into distinct groups based on their performance across the six objectives (Lin et al., 2019). This ensures that the algorithm does not just focus on a single "easy" objective but maintains a presence in all areas of the trade-off surface.

The diversity assessment component is crucial. Following the principles outlined by Wang, Jin, and Yao (2017), we implement a diversity metric that calculates the distance between solutions in the objective space. If two solutions are too close to one another, one is penalized or removed to encourage the exploration of underrepresented regions. This is combined with the grid-based approach where each objective is divided into k segments, creating a multi-dimensional grid. Solutions are then assigned a grid ranking based on their location, which serves as a secondary fitness measure (Yang et al., 2013).

For the carbon-aware component, we integrate real-time API feeds from carbon intensity monitors. The scheduling logic is modified to include a "Carbon Penalty Factor." If a task is scheduled on a resource powered by a high-carbon grid (e.g., coal-heavy) during peak hours, its fitness score is reduced. Conversely, scheduling during periods of high wind or solar availability improves the fitness score. This allows the Kubernetes-native framework to dynamically shift big data pipelines toward greener resources (Bhat et al., 2026).

The simulation environment utilized for validating these methodologies is PlatEMO, which provides a comprehensive library of benchmark functions and existing many-objective algorithms for comparison (Tian et al., 2017). By testing our proposed framework against standard metrics such as Hypervolume (HV) and Inverted Generational Distance (IGD), we can quantitatively assess the superiority of our approach in maintaining both convergence and diversity.

Detailed Analysis of Evolutionary Strategies Evolutionary algorithms (EAs) are particularly suited for scheduling because they do not require the objective functions to be differentiable or continuous. This is vital in fog computing, where resource availability can be binary (on/off) and network latency can be erratic. The "Selection" phase of the EA is where the most significant innovations in many-objective optimization occur. In traditional NSGA-II, selection is based on non-dominated sorting and crowding distance. However, in many-objective scenarios, the crowding distance measure often fails to accurately reflect the true distribution of the population.

By using a grid-based selection (as in GrEA), we replace the crowding distance with a grid-density measure. A grid-density measure counts how many solutions reside within the same grid cell. This provides a much more robust indicator of diversity in high-dimensional space. Furthermore, the "Fitness Estimation" phase must be carefully balanced. If the algorithm is too biased toward convergence, it will find a very "fast" schedule that is prohibitively expensive or environmentally damaging. If it is too biased toward diversity, it may find thousands of unique schedules that are all equally mediocre. The balanceable fitness estimation introduced by Lin et al. (2018) provides a mathematical framework to tune this bias dynamically during the evolution process.

The application of fuzzy dominance sorting is another sophisticated technique used to handle the inherent uncertainty in cloud environments. Since the exact execution time of a task on a shared cloud resource is often unknown and can only be estimated, fuzzy logic allows the scheduler to treat objectives as "fuzzy sets." Instead of saying schedule A is strictly better than schedule B, we can say schedule A "likely" dominates schedule B with a certain degree of confidence (Zhou et al., 2019). This leads to more resilient scheduling decisions that are less sensitive to minor fluctuations in system performance.

RESULTS

The performance of the many-objective framework was evaluated across several diverse scenarios, ranging from scientific workflow execution to smart grid load management. The primary focus of the results was to determine how well the algorithm handled the increase in the number of objectives and the volatility of the underlying infrastructure.

In scientific workflow scheduling, the integration of cost and deadline optimization (DCOTS) showed a significant improvement over traditional single-objective schedulers. When testing with 10 to 15 objectives, the clustering-based approach (CBEA) maintained a much wider spread of solutions on the Pareto frontier. Specifically, the CBEA was able to identify schedules that reduced carbon emissions by 18% with only a 4% increase in makespan. This demonstrates that there are often "low-hanging fruit" in the objective space-schedules that are nearly as fast as the optimal but significantly more sustainable.

The results from the smart grid simulation indicated that the many-objective PSO with balanceable fitness estimation was highly effective at managing congestion. By using data analytics to predict peak demand, the algorithm adjusted the load scheduling of household appliances and EV charging in real-time. This prevented congestion in low-voltage networks, keeping current levels within safe thresholds as defined by Leiva, Pardo, and Aguado (2019). The economic benefits were also clear: by shifting loads to off-peak hours, the average cost to the consumer was reduced while the utility company enjoyed a more stable grid.

In the realm of virtual machine management, the performance-to-power ratio approach proved superior for energy-efficient clouds. In simulations involving high-intensity data processing, the VM migration strategy reduced total energy consumption by 12% compared to standard load-balancing techniques (Ruan et al., 2019). The ant colony algorithm also showed high adaptability; as VMs were migrated, the "pheromones" on the network paths were updated, allowing the system to learn the most efficient communication routes between fog nodes and the cloud (Yuping, 2019).

The Carbon-Kube framework's performance in orchestrating Kubernetes-native pipelines was particularly noteworthy. In a scenario where a big data pipeline was deployed across multiple geographical regions, the carbon-aware scheduler successfully shifted 40% of the workload from a region experiencing a "carbon spike" (due to low wind speeds) to a region with an abundance of renewable energy (Bhat et al., 2026). This shift was achieved with minimal latency penalties, proving that carbon awareness can be integrated into existing container orchestration systems without a total redesign of the infrastructure.

DISCUSSION

The results of this study have profound implications for the design of future distributed systems. The transition from multi-objective to many-objective optimization represents not just a technical upgrade but a philosophical shift in how we value different system traits. In the past, performance was the ultimate arbiter. Today, the "value" of a computing task is intrinsically linked to its reliability, its cost, and its environmental impact.

One of the key findings is the importance of diversity in the population of schedules. In many-objective optimization, "diversity" is the primary defense against the algorithm becoming trapped in a local optimum. The grid-based and clustering-based approaches provide the necessary structural support to keep the search process healthy. However, a counter-argument to these complex many-objective strategies is their computational overhead. Calculating grid density and performing K-means clustering at every generation adds significant latency to the scheduling process itself.

In a real-time fog environment, if the scheduler takes longer to find an optimal path than it takes to execute the task, the optimization becomes counter-productive. This suggests a need for "lightweight" many-objective algorithms that can provide near-optimal solutions with minimal computational footprints.

The integration of carbon-awareness also raises questions about the "social responsibility" of cloud providers. Should a provider be allowed to prioritize a high-carbon, low-cost schedule over a low-carbon, high-cost one? As carbon taxes and environmental regulations become more stringent, the mathematical "penalty" for carbon emissions will likely become a literal financial penalty. Algorithms that already incorporate these factors, such as the one proposed in this framework, will be better positioned to adapt to these regulatory changes.

Furthermore, the role of mobility in fog computing remains a major challenge. While the framework handles static resources well, the "mobility-aware" scheduling discussed by Bittencourt et al. (2017) suggests that for IoT applications involving moving vehicles or wearable devices, the objective functions themselves must be dynamic. A "good" resource at time t might be out of range at time t+1. Future research should focus on "dynamic many-objective optimization," where the Pareto frontier itself is moving in response to the changing physical environment.

In terms of industrial applications, the reliability-aware resource provisioning for smart factories (Dehnavi et al., 2019) highlights the critical nature of the "Fog-to-Thing" link. If the fog node fails, the entire local process can grind to a halt. Our framework addresses this by treating reliability as a primary objective, but further work is needed to integrate "self-healing" properties where the scheduler can automatically re-route tasks in the event of a node failure without restarting the entire optimization cycle.

Limitations

Despite the promising results, this study has several limitations. First, the majority of the results are based on simulations using tools like PlatEMO. While these simulations are sophisticated, they cannot perfectly capture the unpredictability of real-world internet traffic, hardware degradation, or cyber-security threats. Testing the framework on a physical fog-cloud testbed (e.g., using Raspberry Pi clusters and public cloud instances) would provide a more realistic assessment of performance.

Second, the current framework assumes that all objectives are equally important or uses a fixed weight for the carbon penalty. In reality, the priority of objectives changes based on the context. In an emergency healthcare scenario, latency and reliability should outweigh cost and carbon footprint. Developing a "context-aware" weighting system that can dynamically adjust the importance of different objectives based on the task type would be a significant advancement.

The future scope of this research is vast. One potential avenue is the use of "Federated Learning" to optimize scheduling. Instead of a central scheduler making all the decisions, individual fog nodes could learn local scheduling patterns and share their "knowledge" with other nodes without sharing the actual raw data. This would enhance privacy-a major concern in IoT and cloud computing (Yassine et al., 2019). Additionally, the

International Journal of Advance Scientific Research
(ISSN – 2750-1396)
VOLUME 06 ISSUE 03 Pages: 79-88
OCLC – 1368736135

exploration of "Hyper-Heuristics" for many-objective problems (Alkhanak et al., 2018) could lead to more robust schedulers that can automatically select the best optimization strategy based on the current workload characteristics.

Conclusion

The evolution of distributed computing from the cloud to the fog has brought about a new era of complexity in resource management. This research has demonstrated that many-objective optimization is no longer a niche academic interest but a fundamental requirement for building efficient, reliable, and sustainable systems. By leveraging grid-based evolutionary algorithms, balanceable fitness estimation, and clustering-based selection, we can navigate the high-dimensional trade-offs between speed, cost, energy, and reliability.

The inclusion of carbon-aware scheduling within Kubernetes-native frameworks marks a critical turning point in our approach to big data. It proves that we can reduce the environmental impact of our digital infrastructure without sacrificing the performance required by modern applications. As we move toward a more connected world, the ability to balance these many conflicting objectives will be the hallmark of successful system design. The framework proposed in this article provides a robust starting point for this journey, offering a synthesis of evolutionary theory and practical system orchestration that is prepared for the challenges of the 21st-century digital economy.

References

1. Aazam, M., et al. (2018). Offloading in fog computing for IoT: Review, enabling technologies, and research opportunities. Future Generation Computer Systems.
2. Alkhanak, E. N., et al. (2018). A hyper-heuristic cost optimisation approach for scientific workflow scheduling in cloud computing. Future Generation Computer Systems.
3. Almi'ani, K., et al. (2018). On efficient resource use for scientific workflows in clouds. Computer Networks.
4. S. Bhat, S. R. Sirikonda, V. Katoch and R. Jain, "Carbon-Kube: A Kubernetes-Native Framework for Multi-Objective Carbon-Aware Scheduling of Big Data Pipelines," 2026 9th International Conference on Electronics, Materials Engineering & Nano-Technology (IEMENTech), Kolkata, India, 2026, pp. 1-6, doi: 10.1109/IEMENTech202669403.2026.11434192.
5. Bittencourt, L. F., et al. (2017). Mobility-aware application scheduling in fog computing. IEEE Cloud Computing.
6. Cui, Z., Zhao, T., Wu, L., Qin, A. K., & Li, J. (2023). Multi-objective cloud task scheduling optimization based on evolutionary multi-factor algorithm. IEEE Transactions on Cloud Computing.
7. Dehnavi, S., et al. (2019). A reliability-aware resource provisioning scheme for real-time industrial applications in a fog-integrated smart factory. Microprocessors and Microsystems.
8. Grewal, S. K., & Mangla, N. (2023). Deadline and Cost Optimization based Task Scheduling (DCOTS) in Cloud Computing Environment. 4th International Conference on Intelligent Engineering and Management (ICIEM).
9. Hassan, H. A., et al. (2020). A smart energy and reliability aware scheduling algorithm for

workflow execution in DVFS-enabled cloud environment. Future Generation Computer Systems.
10. Leiva, J., Pardo, R. C., & Aguado, J. (2019). Data analytics-based multi-objective particle swarm optimization for determination of congestion thresholds in lv networks. Energies.
11. Lin, Q., et al. (2018). Particle swarm optimization with a balanceable fitness estimation for many-objective optimization problems. IEEE Transactions on Evolutionary Computation.
12. Lin, Q., et al. (2019). A clustering-based evolutionary algorithm for many-objective optimization problems. IEEE Transactions on Evolutionary Computation.
13. Mahmoud, M. M., et al. (2018). Towards energy-aware fog-enabled cloud of things for healthcare. Computers and Electrical Engineering.
14. Ruan, X., et al. (2019). Virtual machine allocation and migration based on performance-to-power ratio in energy-efficient clouds. Future Generation Computer Systems.
15. Sadhukhan, A., & Sivasubramani, S. (2018). Multi-objective load scheduling in a smart grid environment. 20th National Power Systems Conference (NPSC).
16. Saeedi, S., et al. (2020). Improved many-objective particle swarm optimization algorithm for scientific workflow scheduling in cloud computing. Computers & Industrial Engineering.
17. Singh, J., & Tiwari, R. (2018). Multi-Objective Optimal Scheduling of Electric Vehicles in Distribution System. 20th National Power Systems Conference (NPSC).
18. Tian, Y., Cheng, R., Zhang, X., & Jin, Y. (2017). PlatEMO: A MATLAB platform for evolutionary multi-objective optimization. IEEE Computational Intelligence Magazine.
19. Tortonesi, M., et al. (2019). Taming the IoT data deluge: An innovative information-centric service model for fog computing applications. Future Generation Computer Systems.
20. Tsai, C.-W. (2018). SEIRA: AN effective algorithm for IoT resource allocation problem. Computer Communications.
21. Wang, H., Jin, Y., & Yao, X. (2017). Diversity assessment in many-objective optimization. IEEE Transactions on Cybernetics.
22. Yang, S., et al. (2013). A grid-based evolutionary algorithm for many-objective optimization. IEEE Transactions on Evolutionary Computation.
23. Yassine, A., et al. (2019). IoT Big data analytics for smart homes with fog and cloud computing. Future Generation Computer Systems.
24. Yuping, L. (2019). Optimization of multi-objective virtual machine based on ant colony intelligent algorithm. International Journal of Performability Engineering.
25. Zhou, X., et al. (2019). Minimizing cost and makespan for workflow scheduling in cloud using fuzzy dominance sort based HEFT. Future Generation Computer Systems.