MINISTRY OF SCIENCE AND HIGHER EDUCATION OF THE RUSSIAN FEDERATION
Federal State Autonomous Educational Institution of Higher Education
Peter the Great St. Petersburg Polytechnic University

Institute of Computer Science and Cybersecurity

Higher School of Artificial Intelligence Technology

Direction 02.03.01 Mathematics and computer Science

Literature Review

Workflow scheduling algorithms for HPC and cloud environments

Student,
group 5130201/20102 _______________ Gaar V. S.

Supervisor, Ph. D. _______________ Motorin D. E.

«_____»_____________ 2024г.

Saint-Petersburg, 2024

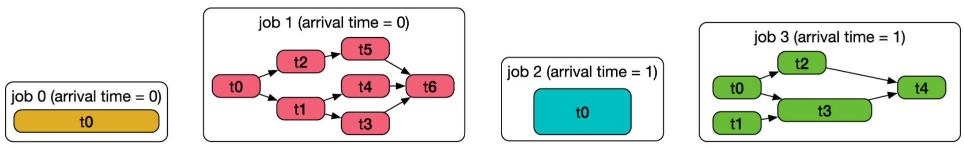
Keywords

Workflow scheduling, HPC, MEC, cloud, DAG, tasks, jobs, DC, privacy.

Introduction

Workflow scheduling plays a pivotal role in modern computing environments, where the complexity and scale of tasks have grown exponentially. These environments include High-Performance Computing (HPC) [1], Mobile Edge Computing (MEC) [2], [6], Edge Function as a Service (Edge FaaS) [4], Geographically Distributed Cloud Data Centers (GD-CDCs) [5], [7], and hybrid cloud-edge ecosystems [8], [9], each characterized by unique challenges and requirements. The surge in data-intensive applications – ranging from artificial intelligence workloads to real-time IoT systems – has amplified the demand for efficient scheduling mechanisms that can dynamically allocate resources, optimize execution time, and minimize energy consumption while adhering to evolving privacy regulations.

The advent of distributed systems introduces complexities that traditional scheduling methods struggle to address. Distributed computing environments are inherently heterogeneous, involving diverse resource types such as CPUs, GPUs, memory, and network bandwidth, which must be coordinated to meet the demands of complex workflows. A common representation of such workflows is through a Directed Acyclic Graph (DAG), as shown in Fig. 1 of [1], where nodes represent tasks, and edges define dependencies. Moreover, the dynamic nature of workloads, where task priorities and resource availability fluctuate unpredictably, exacerbates the need for adaptable and resilient scheduling mechanisms [3]. Additionally, these systems often span multiple geographic regions, as seen in GD-CDCs and hybrid cloud models, where data transfer costs, latency, and privacy constraints further complicate resource allocation [7], [9].



(a) 1 task. (b) 7 tasks. (c) 1 task. (d) 5 tasks.

Fig 1. Examples of tasks and jobs with distinct arrival times and demands (in terms of processing and expected walltime). Each rectangle represents a task: processing is denoted by the rectangle height and execution time by the base's length. The arrows represent the execution's workflow.

Energy efficiency has emerged as a critical consideration, particularly in edge and cloud computing environments, where power consumption directly impacts operational costs and sustainability goals. Studies such as exploration of DVFS-based optimization [5] highlight the growing emphasis on energy-aware scheduling, which seeks to balance resource utilization with energy savings. Similarly, edge-focused systems like those addressed in [3] aim to optimize energy consumption while ensuring low latency, which is critical for real-time applications such as autonomous vehicles and industrial IoT.

Another dimension of complexity lies in the integration of data privacy requirements, particularly in geo-distributed systems. As workflows increasingly operate across multiple jurisdictions, privacy-preserving mechanisms, such as those proposed in [7], are essential to comply with regional data protection laws, reflected in the Fig. 2 of [7] while maintaining operational efficiency. These privacy constraints introduce new trade-offs between WAN usage, data locality, and computational overhead, necessitating innovative approaches to graph partitioning and workflow scheduling [8].

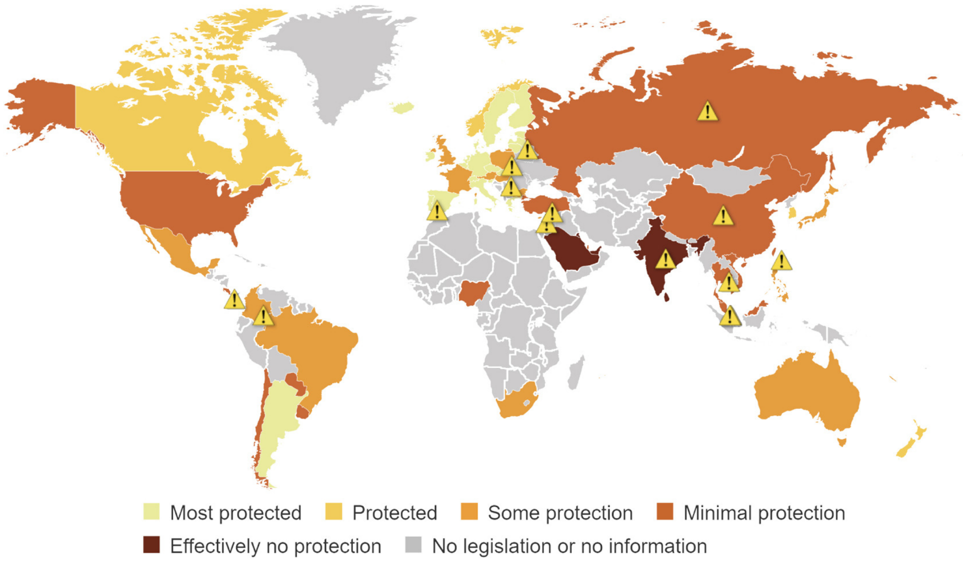


Fig 2. A comparative ranking of 54 countries' privacy and data protection requirements.

The core challenges shared across all ten reviewed studies include:

- Efficient resource allocation in heterogeneous environments, ensuring fair distribution of computational resources such as CPU, memory, and network bandwidth.
- Dynamic task scheduling to adapt to unpredictable workloads and fluctuating resource availability in real time.
- Optimization of key performance metrics, such as execution time, cost, energy consumption, and latency, to achieve global system efficiency.
- Scalability and flexibility, enabling systems to handle growing workloads and infrastructure expansions without significant performance degradation.
- Integration of privacy and regulatory constraints, particularly in systems spanning multiple regions with varying legal requirements [7], [10].

To address these challenges, researchers have turned to machine learning, heuristic and metaheuristic optimization algorithms, and hybrid approaches that combine static and dynamic scheduling strategies. Examples include the use of Actor-Critic Reinforcement Learning [1], multi-strategy heuristic optimization [3], and genetically modified particle swarm optimization [10]. These methods represent a shift towards more adaptable, intelligent scheduling frameworks that balance multiple objectives simultaneously.

This literature review focuses on the comparative analysis of these innovative scheduling approaches, highlighting their strengths, limitations, and potential for integration into future systems. The studies collectively advance the field by addressing the pressing need for scalable, energy-efficient, and privacy-compliant workflow scheduling, offering insights into how these methods can be adapted for the evolving landscape of distributed computing.

1 General analysis

An Actor-Critic RL approach was used in [1], designed to reduce task delays in high-performance computing environments. By framing task scheduling as a directed acyclic graph (DAG), this method allows adaptive queue management, significantly outperforming traditional First-Come, First-Served (FCFS) and Shortest Processing Time (SPT) methods, particularly in terms of throughput. However, its focus on HPC makes it less suited for environments with dynamic workloads or stringent privacy constraints.

In comparison, Feedback Artificial Remora Optimization (FARO) was applied in [2] for MEC environments, balancing CPU, memory, and security requirements. FARO shows high efficiency, achieving low CPU and memory usage (0.012 and 0.010, respectively), which is especially beneficial for mobile edge scenarios. Unlike the method in [1], FARO highlights resource security as a core feature, providing a hybrid optimization approach for security-sensitive workflows where resource availability varies dynamically.

The Multi-Strategy Improved Sand Cat Optimization Algorithm (MSISCSOA), introduced in [3], focuses on reducing delay and energy consumption in heterogeneous edge computing environments. With energy consumption decreased by approximately 19.56%, MSISCSOA emphasizes task adaptability through dynamic search strategies. This method contrasts with FARO's emphasis on security [2], instead optimizing energy efficiency and latency in resource-constrained edge networks. The method's edge-specific design yields lower delay compared to general algorithms, aligning well with latency-critical applications.

In EFaaS environments, a serverless scheduling mechanism combining Highest Bid First and Warm Function First (HBFM and WFFM) is proposed in [4] to enhance execution time. The mechanism is unique in its prioritization and bidding-based resource allocation, achieving efficient multi-user task distribution. Results indicate a decrease in workflow execution time, but unlike the MSISCSOA approach [3], this method is less effective for energy optimization, as it prioritizes task execution time over energy or CPU considerations in serverless environments.

Electricity Price and Energy-Efficient (EPEE) Scheduling, presented in [5], is particularly notable for its energy cost reduction in geographically distributed data centers, where energy consumption varies by location. Leveraging Dynamic Voltage and Frequency Scaling (DVFS), EPEE significantly cuts down energy costs, aligning with the efficiency goals of MSISCSOA [3] but extending them to cloud environments with geographic data distribution. Unlike edge-specific approaches, EPEE accommodates varying electricity prices, making it ideal for distributed cloud systems where energy cost control is crucial.

The method in [6] further innovates within MEC by combining a Marine Predator Algorithm (OMPA) with workload prediction via artificial neural networks (ANNs). The OMPA's unique opposition-based learning prevents local minima, optimizing task scheduling even under fluctuating workloads. Compared to the MSISCSOA algorithm [3], which also targets energy and delay, the method presented in [6] offers superior deadline compliance and VM usage reduction by dynamically predicting workloads, addressing unpredictability in MEC with high efficiency.

Furthermore, [7] focuses on privacy in geo-distributed data centers through Privacy-Preserving Partitioning-based Scheduling (PPPS). Their results show remarkable reductions in WAN usage (up to 99%) and execution time (up to 93%) by addressing complex multi-level privacy constraints. Unlike other studies focused primarily on resource and energy efficiency [1], [3]–[6], [8]–[10], PPPS uniquely addresses regulatory requirements in data privacy. This feature sets it apart as a solution for environments where data transfer is restricted by privacy laws, particularly in scientific workflows operating across multiple jurisdictions.

The Multi-Resource Scheduling Algorithm (MRSA), introduced in [8], focuses on optimizing moldable workflows in HPC systems. By allowing resource reallocation before task execution, MRSA provides a high degree of flexibility, making it particularly effective in managing heterogeneous resources like CPU, memory, and I/O. Unlike Actor-Critic RL used in [1] or FARO in [2], MRSA employs a heuristic-based optimization strategy tailored to HPC environments. This emphasis on multi-resource scheduling aligns conceptually with the multi-strategy optimization in [3] but extends it to include HPC-specific considerations.

Hybrid Scheduling for Hybrid Clouds (HSHC), presented in [9], combines static genetic algorithms with dynamic adjustments to handle hybrid cloud workflows effectively. HSHC is particularly notable for its data locality optimization, which minimizes data transfer costs across cloud environments. This method shares similarities with EPEE [5], which also targets geo-distributed systems but focuses on energy consumption rather than data locality. Additionally, the two-phase structure of HSHC resembles PPPS [7], where distinct optimization stages address different aspects of workflow scheduling.

The Genetically-Modified Multi-Objective Particle Swarm Optimization (GMPSO) algorithm, proposed in [10], integrates genetic operations into PSO for optimizing cost and makespan in hybrid cloud systems. This approach mirrors multi-objective optimization strategies seen in MSISCSOA [3] and FARO [2] but differentiates itself with its unique matrix coding of tasks and resources, offering more granular control over workflow execution. Compared to the heuristic-based MRSA [8] or dynamic HSHC [9], GMPSO is better suited for scenarios requiring simultaneous optimization of multiple objectives.

2 Results

The ten reviewed studies exhibit significant advancements in workflow scheduling, each tailored to address specific challenges within diverse computing environments. This section analyzes the similarities and differences in their results, highlighting shared achievements, unique strengths, and limitations.

2.1 Execution Time Optimization

All studies emphasize minimizing execution time as a primary objective, yet their methods and results vary depending on the targeted environment and algorithmic approach:

- Algorithm in the [1] achieve a 35% improvement in DAG processing speed, demonstrating the effectiveness of Actor-Critic RL in prioritizing task dependencies in HPC environments. This result parallels the performance of MRSA in [8], which also focuses on HPC workflows but achieves optimization by dynamically reallocating resources before execution.
- Similarly, HSHC in [9] reduces execution time by 25% in hybrid cloud environments

through a combination of genetic algorithms and dynamic scheduling, while GMPSO in [10] balances execution time with cost, offering superior performance for hybrid systems.

- Edge computing approaches, such as MSISCSOA in [3], reduce task delay by 21.38%, emphasizing latency-critical applications like IoT. In contrast, OMPA in [6] reduces missed deadlines, improving response times for mobile edge systems.

Similarities:

- A universal focus on reducing makespan and task delays.
- All algorithms optimize task dependencies, whether in DAG-based workflows, used in [1], or moldable workflows in [8].

Differences:

- Methods such as PPPS, presented in [7], while achieving a 93% reduction in execution time, also prioritize privacy, showing that execution time optimization is often coupled with other objectives.

2.2 Energy Efficiency

Energy optimization emerges as a critical concern, particularly for edge and cloud systems:

- EPEE, presented in [5], stands out with significant reductions in energy consumption, leveraging DVFS to adapt workloads across geographically distributed data centers. This focus on energy efficiency is echoed by MSISCSOA in [3], which reduces energy use by 19.56%, balancing it with delay optimization.
- The marine-predator-based approach in OMPA, used in [6] introduces innovative methods for minimizing energy use while ensuring optimal workload distribution, aligning with the energy-saving principles of FARO in [2].

Similarities:

- Both edge (e.g., MSISCSOA in [3]) and cloud systems (e.g., EPEE in [5]) employ heuristic methods to balance energy and resource utilization.

Differences:

- Energy savings in edge systems, such as in [3], focus on reducing latency-related power usage, whereas cloud-centric approaches like in [5] emphasize operational cost savings tied to energy tariffs.

2.3 Cost Optimization

Cost reduction is another recurring objective, especially in hybrid and cloud environments:

- HSHC, presented in [9], achieves a 40% reduction in workflow costs, demonstrating the value of hybrid approaches that adapt resource allocations dynamically based on workload changes.
- GMPSO, used in [10], similarly balances cost and makespan, leveraging genetic enhancements to outperform traditional PSO and heuristic algorithms.

Similarities:

- Cost optimization is a shared focus in cloud systems (e.g., HSHC in [9] and GMPSO in [10]) and MEC environments (e.g., FARO in [2]), highlighting the relevance of balancing resource use with financial constraints.

Differences:

- Privacy-aware systems, such as PPPS in [7], address cost indirectly through WAN usage reduction rather than explicit cost optimization.

2.4 Privacy and Data Locality

Privacy preservation is a unique dimension, primarily addressed in PPPS approach in [7], which minimizes WAN usage by 99% while ensuring compliance with data privacy regulations. This trade-off between performance and privacy is distinct from the goals of other studies, which do not explicitly address data protection.

- However, HSHC, presented in [9], and FARO in [2] share similarities with PPPS, used in [7], in their focus on data locality, reducing data transfer costs while optimizing task allocation.

Similarities:

- Data locality optimization is a shared focus, particularly in hybrid cloud environments (e.g., HSHC in [9]).

Differences:

- Privacy-specific objectives, such as those in [7], highlight a unique focus that is not present in energy- or cost-driven methods.

2.5 Resource Utilization and Scalability

Efficient resource utilization is central to all studies, yet the methods vary significantly:

- MRSA in [8] optimizes multi-resource workflows by reallocating CPU, memory, and I/O dynamically, showing scalability in HPC environments.
- Similarly, EFaaS, used in [4], dynamically prioritizes tasks using a bidding mechanism, ensuring fairness in multi-user environments.

Similarities:

- Dynamic resource allocation is universally employed to address workload variability.

Differences:

- Resource-specific optimizations, such as multi-resource allocation in [8], differ from privacy-focused allocations in [7] or energy-aware distributions in [5].

Conclusion

This review highlights the strengths and limitations of each method, demonstrating the diversity in workflow scheduling needs across different computational architectures. Each method reflects a distinct priority – whether privacy, energy efficiency, latency, or resource optimization – tailored to specific operational environments and regulatory constraints.

Each study's results demonstrate tangible improvements over traditional methods, suggesting that the future of workflow scheduling lies in the integration of machine learning, optimization, and heuristic approaches to handle the growing complexity and scale of distributed computing systems.

References

[1] G. P Koslovski, K. Pereira, and P. R. Albuquerque, "DAG-based workflows scheduling using Actor–Critic Deep Reinforcement Learning," *Future Generation Computer Systems*, vol. 150, pp. 354-363, Jan. 2024, doi: 10.1016/j.future.2023.09.018.

[2] D. K. Sajnani, X. Li, and A. R. Mahesar, "Secure workflow scheduling algorithm utilizing hybrid optimization in mobile edge computing environments," *Computer Communications*, vols. 226–227, Art. no. 107929, Aug. 2024, doi: 10.1016/j.comcom.2024.107929.

[3] P. Jayalakshmi, S. S. Subashka Ramesh, "Multi-strategy improved sand cat optimization algorithm-based workflow scheduling mechanism for heterogeneous edge computing environment," *Sustainable Computing: Informatics and Systems*, vol. 43, Art. no. 101014, Sep. 2024, doi: 10.1016/j.suscom.2024.101014.

[4] S. H. Mahdizadeh, S. Abrishami, "An assignment mechanism for workflow scheduling in Function as a Service edge environment," *Future Generation Computer Systems*, vol. 157, pp. 543-557, Aug. 2024, doi: 10.1016/j.future.2024.04.003.

[5] M. Hussain, L.-F. Wei, A. Rehman, A. Hussain, M. Ali, and M. H. Javed, "An electricity price and energy-efficient workflow scheduling in geographically distributed cloud data centers," *Journal of King Saud University - Computer and Information Sciences*, vol. 36, no. 8, Oct. 2024, doi: 10.1016/j.jksuci.2024.102170.

[6] F. Kuang, Z. Xu, and M. Masdari, "Multi-workflow scheduling and resource provisioning in Mobile Edge Computing using opposition-based Marine-Predator Algorithm," *Pervasive and Mobile Computing*, vol. 87, Art. no. 101715, Dec. 2022, doi: 10.1016/j.pmcj.2022.101715.

[7] Y. Xiao, A. C. Zhou, X. Yang, and B. He, "Privacy-preserving workflow scheduling in geo-distributed data centers," *Future Generation Computer Systems*, vol. 130, pp. 46-58, May 2022, doi: 10.1016/j.future.2021.12.004.

[8] L. Perotin, S. Kandaswamy, H. Sun, and P. Raghavan, "Multi-resource scheduling of moldable workflows," *Journal of Parallel and Distributed Computing*, vol. 184, Art. no. 104792, Feb. 2024, doi: 10.1016/j.jpdc.2023.104792

[9] A. Pasdar, Y. C. Lee, and K. Almi'ani, "Hybrid scheduling for scientific workflows on hybrid clouds," *Computer Networks*, vol. 181, Art. no. 107438, Nov. 2020, doi: 10.1016/j.comnet.2020.107438

[10] H. Hafsi, H. Gharsellaoui, and S. Bouamama, "Genetically-modified Multi-objective Particle Swarm Optimization approach for high-performance computing workflow scheduling," *Applied Soft Computing*, vol. 122, Art. no. 108791, Jun. 2022, doi: 10.1016/j.asoc.2022.108791

Application A. Table comparison of methods

Table 1. Comparison of Methods

Research paper	**Computing environ-ment**	**Scheduling method**	**Optimiza-tion algorithm**	**Objective Function**	**Initial Data**	**Key method features**
DAG-based workflows scheduling using Actor–Critic Deep Reinforce-ment Learning [1]	Data Center (HPC)	Actor-Critic RL	Deep Reinforce-ment Learning (DRL)	Minimize task delay and maximize efficiency	HPC resources as DAG	Deep learning for adaptive queue management policies
Secure workflow scheduling algorithm utilizing hybrid optimization in mobile edge computing environments [2]	Mobile Edge Computing (MEC)	Feedback Artificial Tree (FAT)	Remora Optimization Algorithm (ROA)	Optimize CPU, memory, encryption time	MEC resources and tasks	Hybrid approach for enhanced security and efficiency
Multi-strategy improved sand cat optimization algorithm-based workflow scheduling mechanism for heteroge-neous edge computing environment [3]	Hybrid Cloud-Edge Computing	Sand Cat Optimization (SCOA)	MSISCSOA (Heuristic Swarm Optimization)	Minimize delay and energy consumption	Edge computing resources	Multi-strategy approach with dynamic search
An assignment mechanism for workflow scheduling in Function as a Service edge environment [4]	Edge Function as a Service (FaaS)	Highest Bid First (HBFM), Warm Function First (WFFM)	Bidding + Priority Mechanisms	Minimize makespan	EFaaS resources	Bidding-based and priority assignment mechanisms
An electricity price and energy-efficient workflow scheduling in geographi cally distributed cloud data centers [5]	Geo-distributed cloud data centers	Task ranking and data center selection	Dynamic Voltage and Frequency Scaling (DVFS)	Minimize electricity costs	Geo-distributed cloud data	Uses DVFS and variable energy tariffs
Multi-workflow scheduling and resource provisioning in Mobile Edge Computing using opposition-based Marine-Predator Algorithm [6]	Mobile Edge Computing (MEC)	Marine Predator Algorithm (MPA)	Opposition-based Marine Predator Algorithm (OMPA)	Reduce missed deadlines, minimize VMs	Historical IoT data in MEC	Uses opposition-based learning to avoid local minima
Privacy-preserving workflow scheduling in geo-distributed data centers [7]	Geo-distributed DCs	Privacy-Preserving Graph Partitioning	Privacy-Aware Refinement	WAN usage and privacy adherence	Geo-distributed DCs with privacy levels	Two-stage privacy-preserving workflow scheduling
Multi-resource scheduling of moldable workflows [8]	HPC systems	Multi-resource optimization	MRSA (Resource-aware Scheduling)	Reduce makespan while preserving data privacy	Moldable workflows	Enables pre-execution resource adjustments
Hybrid scheduling for scientific workflows on hybrid clouds [9]	Hybrid clouds	Two-phase Scheduling (Static/ Dynamic)	HSHC (Genetic + Dynamic Adjustment)	Reduce cost, improve time	Scientific workflows	Handles data locality dynamically
Genetically-modified Multi-objective Particle Swarm Optimization approach for high-performance computing workflow scheduling [10]	Hybrid cloud + HPC	Task Mapping via Matrix Encoding	GMPSO (Genetic + PSO)	Optimize cost and makespan	HPC workflows	Introduces genetic operations into PSO

Application B. Table comparison of results

Table 2. Comparison of Results

Research paper	**Perfomance metrics**	**Key result**	**Experimental Data**
DAG-based workflows scheduling using Actor–Critic Deep Reinforce-ment Learning [1]	Makespan, latency, throughput	Improved over FCFS and SPT methods	HPC simulations, DAG graphs
Secure workflow scheduling algorithm utilizing hybrid optimization in mobile edge computing environments [2]	CPU, memory, encryption, security	Reduction in CPU (0.012), memory (0.010)	MEC simulations and real data
Multi-strategy improved sand cat optimization algorithm-based workflow scheduling mechanism for heterogeneous edge computing environment [3]	Delay, energy consumption	Reduced delay by 21.38%, energy by 19.56%	iFogSim testing
An assignment mechanism for workflow scheduling in Function as a Service edge environment [4]	Makespan, resource utilization	Efficient resource allocation in multi-user EFaaS	EFaaS simulation with task prioritization
An electricity price and energy-efficient workflow scheduling in geographically distributed cloud data centers [5]	Energy cost, resource use	Significant reduction in energy costs	Simulations in geo-distributed cloud (CloudSim)
Multi-workflow scheduling and resource provisioning in Mobile Edge Computing using opposition-based Marine-Predator Algorithm [6]	Missed deadlines, VMs	Reduction in missed deadlines and VMs required	iFogSim with NASA and Saskatchewan data
Privacy-preserving workflow scheduling in geo-distributed data centers [7]	Makespan, WAN usage, Security	Reduced makespan by 93%, WAN usage by 99% while preserving privacy constraints	Windows Azure simulation with real-world data
Multi-resource scheduling of moldable workflows [8]	Makespan, Resource Allocation Efficiency	Improved makespan for HPC workflows	HPC simulations with moldable workflows
Hybrid scheduling for scientific workflows on hybrid clouds [9]	Cost Reduction, Makespan	40% cost reduction and 25% faster execution	Hybrid cloud (real-world + simulated workflows)
Genetically-modified Multi-objective Particle Swarm Optimization approach for high-performance computing workflow scheduling [10]	Cost and Makespan Balancing	Superior cost and makespan optimization	Simulated hybrid workflows (CloudSim)