A Systematic Survey and Taxonomy of Energy-Efficient Workflow Allocation Techniques in Cloud Computing Environments

Mazhar Nezami*, Anoop Kumar
Department of Computer Science, College of Computing and Mathematics, Banasthali Vidyapith, Rajasthan, India
E-mail: mazhar.nezami@gmail.com, anupbhola@banasthali.in
* Corresponding author

Keywords: cloud computing, energy efficient, workflow allocation, quality of service, makespan and cost.

Received: September 4, 2025

The cloud computing has revolutionized the way organizations manage and store data. The cloud-based services allow businesses and individuals to access computing resources over the Internet instead of using local servers. As a result of this shift, flexibility, scalability, and cost-efficiency have been enhanced, however, energy consumption has increased. As cloud computing grows, its environmental impact also increases. The cloud infrastructure requires massive amounts of electricity supply, cooling devices, etc. Therefore, energy consumption is one of the primary concerns of cloud computing researchers. In this context, energy-efficient workflow allocation means allocating tasks to virtual machines (VMs) as efficiently as possible in order to save energy, reduce time to complete tasks and lower costs. Since cloud services operate on a pay-per-use model, where users pay based on their usage, optimizing these factors directly benefits cloud providers and users. A systematic literature review (SLR) analyzed 49 studies published from 2015 to 2024, chosen from an initial pool of 585 papers in major academic databases. In this study, a comprehensive taxonomy for energy-efficient workflow allocation (EWA) in cloud computing. It categorizes models by environment (single or multicloud), workflow type (scientific or random), allocation approach (heuristic, meta-heuristic, or hybrid), workload type (static or dynamic), and quality of service (QoS) objectives and constraints. The quantitative analysis shows that 33% of studies used meta-heuristics, 39% used heuristics, and 28% used hybrid approaches. The most common optimization objectives were energy consumption (28%), monetary cost (23%), and makespan (27%). Deadlines (46%) were the most frequently addressed quality of service (QoS) constraint. This study helps researchers in selecting effective energy-efficient workflow allocation (EWA) strategies and highlights open issues, challenges, and future research directions. It serves as a valuable reference for those investigating energy efficiency in cloud computing environments.

Povzetek:

1 Introduction

Cloud computing has transformed the usage of digital resources by enabling on-demand access to services like computing power, storage, and networking. The multi-server model offers flexibility, scalability, and cost efficiency. Users benefit from robust systems without the need for ownership or maintenance. Cloud data centers dynamically allocate computing resources in response to fluctuations in user demand. There are four cloud deployment models: public, private, community, and hybrid. The public clouds like Amazon Web Services (AWS) and Google Cloud Platform (GCP), operate on a pay-as-you-go basis. The private clouds are dedicated to a single organization, offering enhanced control and data security. The community clouds are designed for groups of organizations with shared objectives or regulatory requirements, and hybrid clouds integrate private and public resources, enabling organizations to balance security with operational flexibility. Cloud computing has three main service layers. Infrastructure as a Service (IaaS) gives organizations access to virtual computing resources like processing power, memory, and storage, so they can set up and manage virtual machines. Platform as a Service (PaaS) offers a ready-made environment for building and launching applications, so users do not have to handle hardware or middleware. Examples are Google App Engine and Microsoft Azure. Software as a Service (SaaS) delivers cloud-based apps, such as Google Workspace and Dropbox, which users can access through web browsers without installing anything or doing system maintenance. This setup makes it easier to use and lets people connect from any device with internet access.

A workflow-based application consists of several connected tasks, where each task depends on the successful completion of others. These workflows are used in many areas, including multi-tier web systems, large-scale data analytics, and scientific research. They are also common in fields like healthcare, project scheduling, logistics, image analysis, fake news detection, and genome sequencing. Other examples include traffic forecasting, emotion recognition from speech, facial detection, recommendation systems, license plate identification, and seismic data analysis. In cloud computing, it is important to allocate resources efficiently so tasks finish on time and data centers use less power. This approach helps save energy, make better use of resources, lower costs, and meet Quality of Service (QoS) goals like reliability and cost efficiency.

The main goal of EWA is to assign workflow tasks efficiently to available virtual machines (VMs) while meeting key Quality of Service (QoS) goals such as lowering energy use, meeting deadlines, and cutting costs. As demand for green cloud computing grows, energy-efficient workflow allocation becomes more important for reducing the environmental impact of data centers and meeting user needs. This survey categorizes EWA models based on environment, workload application and type, services, allocation approaches, objectives, and constraints. The primary aim is to investigate energy-related challenges in workflow allocation problems through a comprehensive review of state-of-the-art EWA techniques. The major contributions are as follows:

- It presents a detailed classification of workflow allocation components based on their main features.
- It describes the taxonomy of challenges in energy-efficient workflow allocation, considering allocation strategies, application categories, workload types, and QoS goals and constraints.
- It provides a thorough review of current methods and approaches for energy-efficient workflow allocation in cloud environments.
- The literature review highlights current issues, challenges, and potential future research directions in EWA models.

The structure of the paper is as follows. Section 2 explains the research methodology, includes a related survey, discusses the research gap, objectives, and questions. Section 3 introduces the taxonomy of EWA components. Section 4 describes the taxonomy of EWA challenges and looks at energy efficiency in cloud environments. Section 5 reviews the existing literature, and Section 6 analyzes this literature from several perspectives. Section 7 identifies open issues and suggests future research directions. Section 8 provides the conclusion.

2 Research methodology

This study uses a Systematic Literature Review (SLR) approach, following the guidelines of Kitchenham and Dieste, to make the process reproducible, transparent, and thorough when identifying and analyzing research on EWAs. The review process includes these steps: reviewing related surveys on EWA models to find research gaps, setting research objectives and questions, choosing sources of information, defining search criteria, applying inclusion and exclusion rules, and selecting studies

2.1 Related survey and research gap

A SLR is a thorough and organized assessment of studies focused on a particular topic or research question. The SLRs are designed to give an unbiased summary of current evidence, highlight gaps in research, and suggest directions for future studies. This section summarizes literature reviews about the EWA in cloud computing, with a comparison shown in Table 1. The earlier surveys described EWA models by algorithm type or optimization goal. This new taxonomy introduces analytical dimensions that reveal, algorithmic strategies impact real-world performance. It discusses aspects like algorithmic family, dataset realism, real-time adaptability, and optimization objectives. These factors together provide a picture of EWA research trends. This study addresses a gap by systematically reviewing 49 EWA studies from 2015 to 2024 and points out open issues such as scalability, cost-awareness, and dynamic multi-workflow optimization to help guide future research in sustainable and adaptive cloud computing.

2.2 Research objective

This work has the following research objectives:

O1: Examine existing models and methods for energy-efficient workflow allocation.

O2: Classify EWA approaches by environment, workflow type, algorithmic strategy, workload, and QoS constraints

O3: Identify underexplored optimization goals and emerging research trends in EWA.

Table 1: A comparative analysis using reviews of related literature

Ref.	Survey Year	Key Concept	Taxonomy	Selection of Paper	Year Covered	Open Issue	Study Limitations
[1]	2015	Survey workflow scheduling issues in cloud using a problem–solution approach.	Yes	Not Specified	Not Specified	Yes	Focus on economic, elastic, and robust scheduling, but not on EWA.
[2]	2016	Survey of heuristic, meta-heuristic and hybrid algorithms.	No	Not Specified	Not Specified	Yes	Focuses on cost and time efficient workflow allocation, with limited attention to EWA.
[3]	2016	Survey various challenges of the cloud resource model, application model, scheduling model, and pricing model.	Yes	Not Specified	Not Specified	Yes	The SLR does not address the strengths and weaknesses, nor does it focus on EWAs.
[4]	2019	A detailed evaluation of WA algorithms across cloud, serverless, and fog environments, highlighting their strengths, weaknesses, and suitability for scientific and enterprise workflows.	Yes	Not Specified	Not Specified	Yes	The survey does not cover the EWAs.
[5]	2020	Categorizes workflow scheduling methods based on the multi-objective optimization algorithms	Yes	Not Specified	Not Specified	Yes	The survey does not cover the EWAs.
[6]	2021	Analysis based on metadata that focuses on structure and relationships within the workflow scheduling research community.	Yes	Specified	2011-2020	Yes	The survey is based on metadata-driven analysis.
[7]	2022	Analysis machine learning and artificial intelligence to predict and respond to traffic fluctuations, reducing delays and energy consumption.	No	Specified	2015-2020	No	The survey did not include meta-heuristics approaches with machine learning algorithms for EWAs.
[8]	2022	Examines EWA methods for multi-objective optimization, focusing on energy efficiency through diverse algorithms, scheduling models, resource strategies, and frameworks across computing environments.	No	Specified	2010-2021	No	The survey did not include open issues related to EWAs.
[9]	2023	Analyze various scheduling methods, including heuristic, meta-heuristic, hybrid, and AI/ML based approaches.	Yes	Specified	2010-2023	Yes	The taxonomy is based on workflow scheduling approaches only.

2.3 Research questions

The formulation of research questions and the assessment of the current state of research on energy-efficient workflow allocation constitute the initial steps in the SLR. The research questions guiding this study are as follows:

- RQ1: Which taxonomy dimensions, such as environment, workflow type, allocation strategy, and QoS constraint, most effectively characterize current EWA research?

- RQ2: Which quality of service (QoS) constraints, including deadline, budget, reliability, and energy, are underrepresented in existing EWA models?
- RQ3: How do existing EWA techniques perform with respect to various optimization objectives, and what patterns or gaps are revealed through comparative analysis?

2.4 Source of information

We searched six major databases—IEEE Xplore, SpringerLink, ScienceDirect, ACM Digital Library, Wiley Online Library, and MDPI—as shown in Table 2. The search covered January 2015 to December 2024 to include recent advances in energy-efficient cloud workflow allocation.

Table 2: Data source

Data Source	URL
Springer	https://link.springer.com/
Science Direct	https://www.sciencedirect.com/
IEEE Xplore	https://ieeexplore.ieee.org/
Wiley	https://onlinelibrary.wiley.com/
ACM	https://dl.acm.org/
MDPI	https://www.mdpi.com/

2.5 Search criteria

We set our initial search criteria according to our research goals, choosing key terms like energy efficient workflow allocation and parameters in cloud and related environments. We used logical operators such as OR and AND to combine these terms into a complete search query. To find relevant papers, we searched several digital libraries. The query we used is shown below:

Search String
((workflow) OR (workflow scheduling) OR (workflow allocation) OR (scientific workflow) OR (energy-efficient) OR (minimization) OR (saving) OR (energy AND cost OR budget) OR (energy AND makespan) AND (budget, cost, parameters, deadline, or constraints) AND((approach) OR (algorithm) OR (technique) OR (method)) AND((cloud) OR (cloud computing) OR (cloud environment) OR (IaaS cloud))

2.6 Inclusion and exclusion criteria

In the initial screening, papers that were unrelated to the study were eliminated based on titles. In the second screening, papers were chosen using keywords. The inclusion criteria were established in accordance with the objectives of the review:

- Only studies published in English were included in the analysis.
- The reviewed literature spans the years 2015 to 2024 and focuses on energy-efficient workflow allocation within cloud computing environments.
- The selected papers were sourced from both conferences and academic journals.

In a similar manner, we developed and applied the following exclusion criteria to exclude unnecessary literature:

- Workshop reports, unreviewed articles, and unfinished projects were excluded from consideration.
- Papers written in languages other than English were excluded.
- Articles that do not specify their data sources were excluded.
- Papers that are not relevant to the search query were excluded.
- Duplicate publications in various digital libraries were manually eliminated to prevent the inclusion of identical results.

2.7 Selection strategy

In this section, the method to choose the relevant papers has been described. Moreover, we have limited our search to only include papers published within the last ten years, from 2015 to 2024, on energy-constrained workflow allocation in cloud computing. As shown in Table 3, we implemented the proposed study through a four-stage paper selection process.

Table 3: Study Selection

Process	**Search**	**Selection**	**Screening**	**Review**
Criteria	**Search String**	**Title**	**Abstract**	**Full Article**
Springer	186	76	45	21
Science Direct	146	52	14	12
IEEE	123	41	11	6
Wiley	46	33	15	4
ACM	62	31	17	4
MDPI	22	13	8	2
Total	585	246	110	49

We began by searching for the selected sources and found 585 documents. Using our selection method, which included exclusion criteria, keywords, titles, and full articles, we reduced this number to 246 papers. After reviewing the abstracts, we chose 110 papers based on their content. Finally, we selected 49 studies from the original 585. Of these, 48 were peer-reviewed journal articles and one was a conference paper published on leading academic platforms. We carefully examined each paper to identify research gaps, set the boundaries of our study, and explain the reasons for our research.

2.8 Quality assurance criterion

Quality fulfilment was assessed using a scale ranging from 0 to 1. Table 4 presents the evaluation standards utilized in this study.

Table 4: Quality assessment criteria

Criterion	Description
QA1	Clear problem definition
QA2	Defined algorithmic framework
QA3	Energy/makespan/cost metrics reported
QA4	Relevance to EWA taxonomy
QA5	Replicability of results

3 Taxonomy of workflow allocation components

The taxonomy of cloud computing workflow allocation components offers a systematic framework for categorizing the elements involved in scheduling and managing activities across virtualized resources. Workflow models represent the automation of complex tasks, which are logically connected by data and control flow dependencies and executed on resources according to predefined rules. A Direct Acyclic Graph (DAG) is utilized to depict workflows. In mathematical terms, DAG is defined as Wf = (T, E), where $T = \{t_i, 1 \leq i \leq \mathrm{N}\}$, is set of tasks of workflow and E is the set of edge characterizes the precedence constraints between tasks. The edge $t_i \rightarrow t_j$,indicates the precedence relation between the tasks t_i and t_j in the DAG[10].

The classifications of workflow allocation (WA) components that are intended to address the research question RQ1 have been covered in this part, as illustrated in figure 1. Here's a breakdown of the major components in such a taxonomy:

3.1 Type of environments

The cloud environment provides networking, computation, and storage services that are scalable, secure, efficient, cost-effective, high-quality, on-demand, responsive, and automatically provisioned.

[11]. The environment is classified into two categories: single cloud providers and multi cloud providers.

- Single Provider Environment: When users rely on a single cloud service provider (CSP), CSP manages all the resources required to run user applications. These applications are deployed on VMs provisioned by a single CSP[12], [13].
- Multi-Provider Environment: A multi-cloud environment uses services from several cloud providers to process computing needs. Instead of depending on a single provider, organizations run applications and workloads on multiple clouds. This approach offers more options, helps manage costs, and provides extra backup. Each provider offers different resources, configurations, and price models. The organizations can choose what best fits each task. However, working with more than one provider can make management and communication harder because teams must coordinate between them.

3.2 Workflow application

A Workflow Management System (WMS) helps make complex workflows easier to manage and automates their execution in computing environments. It keeps tasks in order, so each one runs only after the previous steps are done. Workflow applications are usually grouped as either random or scientific.

- Random workflow: In such workflows do not have a fixed number of task edges or depth levels. This makes them useful for simulating workflow applications.
- Scientific workflow: is used for real-world scientific applications like CyberShake, LIGO, Montage, Epigenomics, and SIPHT. Such workflows might need a lot of data, memory, CPU power, or input/output resources.
-

3.3 Workflow allocation approaches

Workflow applications are dynamic, and when combined with precedence constraints, they become more complex. The QoS factors like energy efficiency, execution time, and cost add further challenges. It is important to optimize schedules while meeting these QoS standards. Researchers have developed several approaches, such as exact methods, heuristics, metaheuristics, and hybrid techniques, to tackle these issues.

- Heuristic: The heuristic approach to workflow automation (WA) in cloud environments seeks to identify feasible solutions by utilizing practical experience and problem-specific characteristics, especially when finding an optimal solution is infeasible due to high complexity. Since WA problems are NP-hard, obtaining optimal solutions for large-scale workflows within a reasonable timeframe is challenging. Therefore, heuristics provide a practical compromise by producing sub-optimal yet efficient solutions, with the principal advantage being reduced time complexity [14], [15], [16] , [17].
- Meta-heuristic: The meta-heuristic algorithms take inspiration from natural processes like annealing, particle swarm optimization, and bee colony optimization to solve complex optimization problems. In cloud environments, they offer efficient ways to allocate workflows, especially when finding exact solutions is too difficult. In comparison to traditional heuristic methods, metaheuristics usually deliver better results in less time[18] , [19], [20] [21], [22].
- Hybrid: The heuristic methods are usually faster, but they often do not find the best solutions for large-scale optimization problems. On the other hand, meta-heuristic algorithms search a wider range of possibilities, although they require more computing power. Combining both methods allow a system to use the speed of heuristics and the broad search ability of meta-heuristics, leading to more balanced and

effective optimization results[12] [23], [24] [25], [26], [27] [28], [29], [30],[31][10], [32].

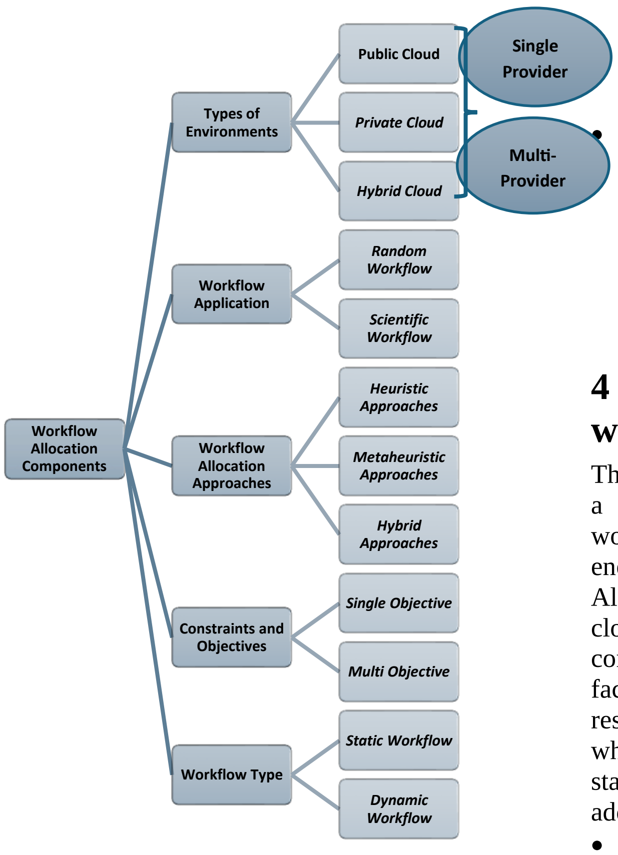

Figure 1: Taxonomy of workflow allocation components

3.4 Workflow allocation objectives and constraints

The following two categories apply to the workflow allocation objectives:

- Single Objective: The aims of single objective optimization is to identify the optimal solution for a function that addresses only one objective, such as cost, makespan, energy, or utilization.
- Multiobjective: In real-world workflow allocation, we often optimize objective functions with multiple goals at the same time. When competing objectives are achieved in multi-objective optimization, there is not a single optimal solution. Here, several compromise solutions known as the trade-off, nondominated, or Pareto-optimal solutions are calculated.

3.5 Workflow types

The following two categories apply to the workflow workload types:

- Static: Static workload allocation refers to a scenario where the characteristics of tasks and resource requirements are known in advance. In such workflow tasks, their execution time, resource requirements, and task dependencies are predefined and do not change during execution. The resources (like VMs, CPU, and storage) are allocated before execution based on the predicted workload.
- Dynamic: To deal with the lack of scheduling data, such as task size, execution time, communication cost, resource capabilities, etc., dynamic scheduling was created. During runtime a specific piece of information is obtained. After that, they allotted cloud resources dynamically at runtime in accordance with certain policies.

4 Taxonomy of energy efficient workflow allocation challenges

This section addresses research question RQ2 and presents a taxonomy of challenges related to energy-efficient workflow allocation. The first subsection identifies key energy requirements for developing an EWA model. Allocating workflow tasks to virtual machines (VMs) in a cloud environment while adhering to SLA and QoS constraints requires careful consideration of multiple factors. The primary objective is to ensure efficient resource utilization to achieve energy consumption targets while maintaining performance and service quality standards. The following detailed requirements must be addressed:

- Task Characteristics: Knowing the execution time of tasks helps allocate resources to minimize idle time and ensure tasks are completed within the expected timeframe.
- Workflow Characteristics: Workflow tasks often have dependencies where some tasks cannot start until others finish. DAG-based modelling of task dependencies helps capture these constraints and enables the use of algorithms such as CPM or DAG-optimized scheduling.
- Energy Efficiency Considerations: Each VM and physical host in the cloud environment consumes energy based on the workload.
- Resource Provisioning and Scheduling Strategy: Efficient allocation involves mapping tasks so that each resource (CPU, memory, and storage) is used optimally.
- Task Execution Time and Makespan Reduction: A shorter makespan leads to quicker completion and the earlier release of resources, thereby indirectly reducing energy consumption.
- CO_2 Emissions: This growing energy consumption contributes significantly to CO_2 emissions, making data centers responsible for around 2% of global greenhouse gas emissions. To create sustainable, eco-friendly data centers, it is crucial to implement strategies that reduce energy consumption and minimize CO_2 emissions.
-

4.1 Quality of service (QoS) challenges

QoS objectives play a key role in determining the efficiency and effectiveness of EWA. These objectives are related to and contribute to the overall performance of cloud-based workflow systems.

4.1.1 Objective challenges

- Makespan: defined as the duration from the initiation of the first task to the completion of the final task.
- Energy Consumption: refers to the total power consumed by computing devices in cloud systems, including RAM, storage disks, and network interfaces. This consumption increases during the execution of workflows on cloud infrastructure.
- Throughput: denotes the number of tasks or workflows successfully completed within a specified time period.
- Resource Utilization: indicates the extent to which allocated resources are used efficiently within the cloud system.
- Response Time: refers to the amount of time it takes for a system to respond to user input. Basically, it refers to the time needed for tasks associated with workflow on cloud resources to be allocated.
- Monetary Cost: The total expense incurred for resource usage is calculated based on the billing interval, typically hourly.
- Speedup: The speedup (SP) is the ratio of the sequential execution time to the schedule's makespan.

4.1.2 Constraints

- Task Dependency Constraints: A workflow is often represented as a DAG [33], where tasks depend on each other. A task cannot start until its preceding task has been completed, since dependencies must be respected.
- Deadline: the time limit for executing the workflow[34]. In connection with makespan as a scheduling goal, deadlines necessitate different decisions from a scheduling system.
- Budget: Each workflow usually has a set budget. The total cost of using cloud resources like virtual machines, storage, and data transfer needs to stay within this limit.
- Reliability and Fault Tolerance Constraints: Cloud environments are susceptible to failures, including virtual machine crashes and network outages[13], [14], [35]. Workflow scheduling may need to incorporate fault tolerance techniques to ensure reliability and prevent task failures.

5 Literature on energy efficient workflow allocation models

In this section, we systematically reviewed the energy-efficient workflow allocation (EWA) model based on classification, as illustrated in figure 1.

Table 5 systematically reviews key EWA models evaluating their strengths and weaknesses. The heuristic approaches viz. CEAS [15], HPEFT[17], DRAWS[16] and REEWS[14] good in low-overhead scheduling and quick convergence, making them efficient for static or commercial multi-cloud workloads with strict deadlines. The meta-heuristic models viz. C-PSO[19], NSGA-II-ELNU[21], IWDC[22], MOGA[20] and ECMSMOO[18] achieve significant improvements in energy efficiency and makespan reduction through global search and adaptive learning. The hybrid model viz. HBMMO[25], GA-PSO[26], HGAABC[27],HGALO-SCA[28], HCGWO[30], HEFT-ACO[29], ALPSO[31] HSMO[23] , EBABC-PF[24] and HAED[12], outperform single-paradigm methods in multi-objective optimization, balancing energy, cost, and load with higher accuracy. Some methods, such as EnReal[36] ,MWSTR[37] and, integrate DVFS and VM-migration strategies, effectively reducing idle power and optimizing utilization under variable load.

Across the surveyed EWA models several recurrent limitations emerge viz. limited scalability, neglect of inter-task communication and data transfer costs, and lack of real-time adaptability. Many algorithms EERS[13], ECMSMOO[18], ALPSO[31], EnReal[36], MWSTR[37], RMREC[38], CAAS[39], EATTO[40], ACRR[41], EM_WOA[42], EVMP[43], SEPSO[60], ELSCiW[61] were validated on small or scientific workflows. Their runtime or convergence performance degrades sharply in large-scale scientific or multi-cloud environments. Several models NSGA-II-ELNU[21], GA-PSO[26], EEWS[44], EATTO[40] treat workflows as computation-centric while ignoring data transfer latency, bandwidth variability, or VM co-location effects. Such limitation cause underestimation of energy and cost in realistic distributed settings, particularly in data-intensive workflows. Most approaches HAED[12] ,CAAS[39], OWS-MRL[45] DCMORL[46] rely on static pre-execution optimization, which limits responsiveness to changing workloads, resource conflict, or energy constraints in heterogeneous cloud systems.

Table 6 systematically categorizes EWA approaches based on multiple parameters, including environment type (single vs. multi-cloud), workflow type, allocation strategy (heuristic, meta-heuristic, hybrid), workload nature (static vs. dynamic), optimization objectives (energy, cost, makespan), and QoS constraints (deadline, budget, reliability). It reveals that prior literature mainly concentrated on cost, makespan and utilizations as the dominant objectives, while factors like energy, reliability, fault tolerance and response time remain underexplored.

Table 7 compares EWA models across a comprehensive set of objectives and constraints.

Acronyms are used in the tables for objectives and constraints, such as:
Makespan (MK),
Resource Utilization (RU),

Convergence Ratio(CR),
Monetary Cost (MC),
Energy Consumption (EC),
Load Balance (LB),
Security (SC),
Response Time (RT),
Fault Tolerance (FT),
Throughput (TP),
Deadline (D),
Budget (B),
Reliability (R),

Acronyms are used in the tables for benchmark workflow, such as:

Montage(M),
SIPHIT(S),
CyberShake(C),
LIGO(L),
Epigenomics(E),
Gaussian Elimination(G),
Fourier Transformation(F),
Random(R).

In most existing models (e.g., CEAS, EnReal, ASFLA, ECMSMOO, C-PSO) strongly emphasize makespan, energy, and cost optimization, indicating that energy–performance trade-offs dominate EWA research. The majority of frameworks fail to consistently implement quality of service (QoS) constraints, including deadline adherence, fault tolerance, and reliability. Consequently, their real-time adaptability remains limited.

Table 5: Energy efficient workflow allocation models

Model	Strength	Weakness
CEAS [15]	Suitable for commercial multi-cloud environments, as it enables energy savings through effective utilization of the gap between makespan and deadline.	CEAS approach employs comprehensive coding strategies to achieve energy savings.
DRAWS[16]	DRAWS dynamically adjusts task priorities in response to changing objective weights.	DRAWS evaluates a limited set of three workflows.
EnReal[36]	Live VM migration from an underutilized physical machine.	Live migrations result in higher memory overhead.
ASFLA[47]	Resources are allocated dynamically, enabling scalable adjustments to meet demand and promote efficient utilization.	Parameters need to be adjusted for best performance.
ECMSMOO[18]	Minimizing makespan, economic cost, and energy consumption.	The performance of large-scale workflow applications are not discussed.
C-PSO[19]	C-PSO demonstrated significant improvements in both makespan and execution cost for large-scale workflows.	C-PSO is susceptible to premature convergence.
MWSTR[37]	The DVFS technique was employed to balance schedule length and energy consumption by reclaiming slack time.	In MWSTR, the performance of large-scale workflows are not considered.
NSGA-II-ELNU[21]	Faster convergence to the Pareto front by simplifying sorting operations.	Inter-task communication costs and dependencies are ignored.
IWDC[22]	IWDS demonstrates greater cost efficiency regardless of the workflow structure.	Performance and the cost of scheduling are influenced by type of VM instance selected.
MOGA[20]	The approach manages both dependent and independent tasks while adhering to user-defined budget and deadline constraints.	In the absence of independent tasks, gaps are unutilized, resulting in poor utilization.
EATS[48]	EATS demonstrates 38% greater energy savings compared to DEWTS and 20.93% higher resource utilization than EES.	EATS shows insignificant performance when the system operates with a small number of processors.
SECPS[49]	VMs are deployed using the shortest path based on energy consumption metrics,	SECPS does not address the performance of large-scale workflow applications.
HBMMO[25]	Search and compute the non-dominated solutions efficiently.	Integration of PEFT with the SOS increases the complexity of the algorithm.
GA-PSO[26]	GA-PSO outperforms GA by 16% (makespan), 13% (cost), 28% (load balance), and PSO by 4% for all metrics.	Inter-task communication costs and dependencies are ignored.

Qureshi B[50]	Energy efficiency improves by 38% over HEFT.	The performance of large-scale workflow applications are not discussed.
EEWS[44]	CPU performance is evaluated based on the time required to complete a task.	The communication cost between tasks is not considered in this analysis.
REEWS[14]	Critical tasks are assigned the highest priority, thereby mitigating starvation of low priority tasks.	Multiple scheduling orders are possible due to the topological arrangement of tasks
HGAABC[27]	The HGAABC algorithm demonstrates superior convergence performance compared to both MABC and MGA.	HGAABC increases the algorithmic complexity.
HPEFT[17]	Execution time reduced by 5% to 16% compared to classical algorithms.	Computing time of HPEFT increases by 50% when layers increases.
JAYA[51]	Common fitness function ensures fair evaluation of all optimization algorithms.	The experimental setup and parameters are unknown.
HAED[12]	HAED outperforms NSGA-II and HPSO with higher hypervolume across workflows.	Hybrid HAED increases the complexity of the algorithm.
SERAS[35]	SERAS achieves up to 96% faster execution and 55% lower energy than HEFT, DEWTS, Wu, and Safari.	SERAS algorithm Overall complexity of $O(n^2)$.
HGALO-SCA[28]	Random chaos helps escape local optima and speeds convergence.	HGALO-SCA does not provide comprehensive assessment.
RMREC[38]	It lowers task data migration, reducing communication energy use.	Performance gains shown for Epigenomics and Gaussian Elimination workflows only.
OWS-MRL[45]	Significant cost and power saving compared to MCP and ETF.	The resources operating at minimum frequency can cause transient errors.
CAAS[39]	Containers use fewer resources as they exclude OS images.	Overhead associated with container management.
ANFIS[52]	Shows higher fault tolerance than IDE and ACO.	Considers only VM faults, ignoring network and I/O reliability.
I_MaOPSO[53]	I_MaOPSO improves Hypervolume by up to 71% over LEAF, 182% over MaOPSO.	Roulette wheel leader selection fails with large or identical population values.
EATTO[40]	EATTO achieves a balanced trade-off among conflicting algorithmic objectives by employing a unified objective function.	Inter-task communication costs and dependencies are ignored.
DCMORL[46]	DCMORL improved execution cost and energy consumption compared to IC-PCPD2, CEAS, S-CEDA and HPSO.	Chebyshev scalarization function is not effective when one objective heavily outweighs the others.
HCGWO[30]	Enhance GWO convergence speed while reducing local optima traps.	Chaos theory adds extra overhead from generating and managing chaotic maps.
EAFSAIPR[54]	Efficiently meets deadlines, cuts execution time, and optimizes budget in task replication.	Task replication and cryptographic operations add extra overhead.
HEFT-ACO[29]	HEFT-ACO remains effective in both small- and large-scale workflow.	The workflow characteristics, viz. balance or asymmetry, are not considered.
EASVMC[55]	WWO enables significant energy savings by maximizing resource utilization and reducing the number of VM migrations.	The complexity of the algorithm is $O(n^2v)$, also the risk of premature convergence during VM consolidation.
EERS[13]	Reduces energy consumption while simultaneously maximizing system reliability.	The reliability model considers only those errors that are influenced by CPU frequency.
ALPSO[31]	It shows high convergence rate, searchability of ALO, and communication capacity of PSO's enhance the algorithm performance.	The characteristics of randomly generated workflow and variability in task numbers are not considered.
PACS[56]	PACS demonstrates superior performance compared to RRA,GA, PSO, and ACO.	The performance of large-scale workflow applications are not discussed.
ACRR[41]	It shows the average execution Time is 31.162 seconds.	Performance improvement is demonstrated only CyberShake

		workflow only.
HSMO[23]	The HSMO algorithm demonstrated superior performance compared to the ABC, PSO.	The integration of SMO and BDSD in the hybrid HSMO increases complexity.
EM_WOA[42]	EM_WOA significantly outperforms both WOA and PSO.	Task prioritization is not considered and small budgets hinder PSO and WOA scheduling .
EVMP[43]	It reduces execution delays by decreasing both transfer time and VM creation time.	Performance improvements are demonstrated only for the Pan-STARRS workflows.
BDCE, BDD[57]	BDCE and BDD achieve highest success rate for both budget and deadline constraint workflow for DVFS and non-DVFS resources.	The simulation task set considered is for medium workflow only.
EBABC-PF[24]	EBABC-PF outperforms HEFT, DHEFT, and NSGA-II by maximizing utilization while reducing makespan and processing cost across all benchmark workflows.	The performance of EBABC-PF can be affected by changing one or more of the parameters.
COSA[58]	By leveraging the global search capability of NSGA-II and the rapid convergence of OSA, COSA achieves an effective balance between exploration and exploitation.	The performance multiobjective optimization of large-scale workflow applications are not discussed.
Choudhary et al.[59]	Clustering techniques significantly reduce data transmission costs .	The algorithm shows high time complexity.
EIS[60]	EIS allocates workflow slack time among tasks based on each task's optimal execution time, conserving energy through voltage and frequency adjustments.	The algorithm exhibits high computational time complexity.
PMWS-HC[61]	MSIA shows superior balance among solution diversity, convergence, execution time, and the number of leased public cloud VMs.	The execution of privacy-sensitive tasks on a private cloud can result in inefficient utilization and increased execution time.
SEPSO[62]	SEPSO dynamic scheduling framework assigns tasks to either private or public cloud resources.	Performance improvements are demonstrated exclusively for the CyberShake and Montage workflows.
ELSCiW[63]	achieves a reduction in energy consumption ranging from 4.71% to 11.19%, and a decrease in latency between 5.35% and 12.92%.	The performance of large-scale workflow applications are not discussed.

Table 6: Classification of EWA Models

Ref	**Year**	**Environment**		**Workflow**				**WA Model**			**Workload Type**	
		S	**M**	**Random**		**Scientific**		**H**	**MH**	**HB**	**Static**	**Dynamic**
				S	**M**	**S**	**M**					
[15]	2015	×	✓	×	×	✓	×	✓	×	×	✓	×
[16]	2015	✓	×	×	×	✓	×	✓	×	×	✓	×
[36]	2016	×	✓	×	×	✓	×	✓	×	×	✓	×
[47]	2016	✓	×	✓	×	✓	×	×	✓	×	✓	×
[18]	2016	✓	×	×	×	✓	×	×	✓	×	✓	×
[19]	2016	✓	×	×	×	✓	×	×	✓	×	✓	×
[37]	2017	✓	×	×	✓	×	×	✓	×	×	✓	×
[21]	2017	✓	×	✓	×	×	×	×	✓	×	✓	×
[22]	2017	✓	×	×	×	✓	×	×	✓	×	✓	×
[20]	2018	✓	×	✓	×	×	×	×	✓	×	✓	×
[48]	2018	✓	×	✓	×	×	×	✓	×	×	✓	×
[49]	2018	✓	×	✓	×	×	×	✓	×	×	✓	×
[25]	2018	✓	×	×	×	✓	×	×	×	✓	✓	×

[26]	2018	✓	×	×	×	✓	×	×	×	✓	✓	×
[50]	2018	✓	×	×	×	✓	×	✓	×	×	✓	×
[44]	2019	✓	×	×	×	✓	×	×	×	✓	✓	×
[14]	2019	✓	×	✓	×	×	×	✓	×	×	✓	×
[27]	2019	✓	×	×	×	×	✓	×	×	✓	✓	×
[17]	2019	✓	×	×	✓	×	×	✓	×	×	✓	×
[51]	2019	✓	×	×	×	✓	×	×	✓	×	✓	×
[12]	2020	✓	×	×	×	✓	×	×	×	✓	✓	×
[35]	2020	✓	×	×	×	✓	×	✓	×	×	✓	×
[28]	2020	✓	×	×	×	✓	×	×	×	✓	✓	×
[38]	2020	✓	×	×	×	✓	×	✓	×	×	✓	×
[45]	2020	✓	×	×	×	✓	×	✓	×	×	✓	×
[39]	2020	✓	×	✓	×	×	×	✓	×	×	✓	×
[52]	2020	✓	×	×	×	✓	×	×	×	✓	✓	×
[53]	2020	✓	×	×	×	✓	×	×	✓	×	✓	×
[40]	2020	✓	×	✓	×	×	×	×	✓	×	✓	×
[46]	2020	✓	×	×	×	✓	×	×	×	✓	×	✓
[30]	2020	✓	×	×	×	✓	×	×	×	✓	✓	×
[54]	2021	✓	×	×	×	✓	×	×	✓	×	✓	×
[29]	2021	×	✓	×	×	✓	×	×	×	✓	✓	×
[55]	2021	✓	×	×	×	✓	×	×	✓	×	✓	×
[13]	2021	✓	×	×	×	✓	×	✓	×	×	✓	×
[31]	2021	✓	×	✓	×	×	×	×	×	✓	✓	×
[56]	2021	✓	×	✓	×	×	×	✓	×	×	✓	×
[41]	2021	✓	×	×	×	✓	×	✓	×	×	✓	×
[23]	2021	✓	×	×	×	✓	×	×	×	✓	✓	×
[42]	2022	✓	×	×	×	✓	×	×	✓	×	✓	×
[43]	2022	✓	×	✓	×	×	×	✓	×	×	✓	×
[57]	2022	✓	×	×	×	✓	×	✓	×	×	✓	×
[24]	2022	✓	×	×	×	✓	×	×	×	✓	✓	×
[58]	2022	✓	×	×	×	×	✓	×	×	✓	✓	×
[59]	2022	✓	×	×	×	✓	×	×	✓	×	✓	×
[60]	2023	✓	×	×	×	✓	×	×	✓	×	✓	×
[61]	2023	×	✓	×	×	×	✓	✓	×	×	✓	×
[62]	2023	×	✓	×	×	×	✓	×	✓	×	✓	×
[63]	2024	✓	×	✓	×	×	×	×	✓	×	✓	×

Table 7: Comparison based on objective, constraints and benchmark workflow of EWA Model

Ref	Objectives										Constraints			Benchmark Workflow						
	MK	RU	CR	MC	EC	LB	SC	RT	FT	TP	D	B	R	M	S	C	L	E	G	F	R
[15]	×	×	×	✓	✓	×	×	×	×	×	✓	×	×	✓	✓	✓	✓	×	×	×	×
[16]	✓	×	×	✓	✓	×	×	×	×	×	✓	×	✓	✓	×	✓	✓	×	×	×	×
[36]	×	✓	×	×	✓	×	×	×	×	×	×	×	×	×	✓	×	×	✓	×	×	×
[47]	×	×	×	✓	×	×	×	×	×	×	✓	×	×	✓	×	✓	✓	×	×	×	✓
[18]	✓	×	×	✓	✓	×	×	×	×	×	×	×	×	✓	×	×	×	✓	×	×	×
[19]	✓	×	×	✓	×	×	×	×	×	×	✓	×	×	✓	×	✓	✓	✓	×	×	×
[37]	✓	×	×	×	✓	×	×	×	×	×	×	×	×	×	×	×	×	×	✓	✓	✓
[21]	✓	×	×	×	✓	×	×	×	×	×	×	×	✓	×	×	×	×	×	✓	✓	✓
[22]	✓	×	×	✓	×	×	×	×	×	×	×	×	×	✓	✓	✓	✓	✓	×	×	×
[20]	✓	×	×	✓	✓	×	×	×	×	×	✓	✓	×	×	×	×	×	×	×	×	✓
[48]	✓	✓	×	×	✓	×	×	×	×	×	✓	×	×	×	×	×	×	×	×	×	✓
[49]	✓	×	×	×	✓	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	√
[25]	✓	✓	×	✓	×	×	×	×	×	×	×	×	×	✓	✓	✓	✓	✓	×	×	×
[26]	✓	×	×	✓	×	✓	×	×	×	×	✓	✓	×	✓	✓	✓	✓	✓	×	×	×
[50]	×	✓	×	×	✓	×	×	×	×	×	×	×	×	×	×	✓	×	✓	×	×	×
[44]	✓	×	×	×	✓	×	×	×	×	×	✓	×	×	✓	✓	✓	✓	✓	×	×	×
[14]	×	×	×	×	✓	×	×	×	×	×	✓	×	✓	×	×	×	×	×	✓	×	✓
[27]	✓	×	×	✓	×	×	×	×	×	×	×	×	×	✓	✓	✓	✓	×	×	×	×
[17]	✓	×	×	×	×	×	×	×	×	×	✓	×	×	×	×	×	×	×	×	×	✓
[51]	✓	×	✓	✓	×	×	×	×	×	×	×	×	×	✓	✓	✓	✓	×	×	×	×
[12]	✓	×	×	✓	✓	×	×	×	×	×	✓	×	×	✓	✓	×	✓	✓	×	×	×
[35]	✓	×	×	×	✓	×	×	×	×	×	✓	×	✓	✓	✓	✓	×	✓	×	×	×
[28]	✓	×	×	✓	✓	×	×	×	×	✓	×	×	×	✓	✓	×	✓	✓	×	×	×
[38]	×	×	×	✓	✓	×	×	×	×	×	×	✓	×	×	×	×	×	✓	✓	×	×
[45]	✓	✓	×	✓	✓	×	×	×	×	×	✓	×	×	×	✓	✓	✓	×	×	×	×
[39]	✓	✓	×	×	✓	×	×	×	✓	✓	✓	×	×	×	×	×	×	×	×	×	✓
[52]	✓	×	×	✓	✓	×	×	×	✓	×	✓	×	×	✓	×	✓	✓	✓	×	×	×
[53]	✓	×	×	✓	✓	×	×	×	×	×	✓	×	✓	✓	×	✓	×	✓	×	×	×
[40]	✓	×	×	×	✓	×	×	×	×	✓	×	×	×	×	×	×	×	×	×	×	✓
[46]	×	×	×	✓	✓	×	×	×	×	×	✓	×	×	✓	✓	✓	×	✓	×	×	×
[30]	✓	×	×	✓	✓	×	×	×	×	×	×	×	×	✓	×	✓	✓	✓	×	×	×
[54]	✓	✓	×	✓	×	×	✓	×	×	×	✓	×	×	✓	✓	✓	✓	×	×	×	×
[29]	✓	×	×	✓	×	×	×	×	×	×	×	×	×	✓	×	✓	✓	×	×	×	×
[55]	×	✓	×	×	✓	×	×	×	×	×	×	×	×	✓	✓	✓	✓	✓	×	×	×
[13]	×	✓	×	×	✓	×	×	×	×	×	×	×	✓	✓	×	✓	×	×	×	×	×

[31]	✓	×	×	✓	✓	✓	✓	×	×	×	×	×	✓	×	×	×	×	×	×	×	✓
[56]	✓	×	×	✓	✓	×	×	✓	×	×	×	×	×	×	×	×	×	×	×	×	✓
[41]	✓	×	×	✓	✓	×	×	×	✓	✓	×	×	✓	×	×	✓	×	×	×	×	×
[23]	✓	×	×	✓	×	×	×	×	×	×	✓	✓	×	✓	✓	✓	✓	✓	×	×	×
[42]	×	×	×	×	✓	×	×	×	×	×	×	✓	×	×	×	×	×	✓	✓	×	×
[43]	✓	✓	×	×	✓	×	×	×	×	×	✓	×	×	×	×	×	×	×	×	×	✓
[57]	✓	✓	×	×	✓	×	×	×	×	×	✓	✓	×	✓	✓	✓	✓	✓	×	×	×
[24]	✓	✓	×	✓	×	×	×	×	×	×	×	×	×	✓	✓	✓	✓	✓	×	×	×
[58]	✓	×	×	✓	✓	×	×	×	×	×	✓	✓	×	✓	×	×	✓	✓	×	×	×
[59]	×	×	×	✓	✓	×	×	×	×	×	✓	×	×	✓	✓	✓	✓	×	×	×	×
[60]	×	×	×	×	✓	×	×	×	×	×	✓	×	×	✓	✓	✓	✓	✓	×	×	×
[61]	×	×	×	✓	✓	×	×	×	×	×	✓	×	×	✓	✓	✓	✓	✓	×	×	×
[62]	✓	✓	×	✓	✓	×	×	×	×	×	✓	✓	×	✓	×	✓	×	×	×	×	×
[63]	×	✓	✓	×	✓	✓	×	×	×	×	✓	×	×	×	×	×	×	×	×	×	✓

Table 8: Comparison based on taxonomy dimensions and research implications

Dimension	Description	Observed Model Range	Research Implication
Algorithmic Family	Heuristic, Meta-heuristic, Hybrid	Clear dominance of heuristic models (≈ 39%)	Emphasizes need for multi-objective trade-off handling
Runtime Complexity	Computational overhead per iteration or population	Heuristic models→ Low Meta-heuristic→ Medium Hybrid / AI- models → High	Hybrid models are computationally heavier; trade-offs with accuracy
Dataset Realism	Type of workflow dataset (Scientific, real)	Mostly Scientific benchmarks (Montage, SIPHIT, CyberShake)	Necessity for real or hybrid IoT datasets
Real-Time Adaptability	Dynamic response capability	Present in ML/RL hybrids (≈ 10%)	Sustainable and intelligent EWA: emerging directions
Optimization Objectives	Energy, Cost, Makespan, Utilization, Reliability	Multi-objective dominance; imbalance across metrics	Integrates energy, reliability, and sustainability goals

Table 8 illustrates the relationship between algorithmic categories and operational contexts. A heuristic model, comprising about 39% of the sample, is the most common because it requires less computational power.

The balance between exploration and exploitation is achieved with metaheuristics, but their sensitivity to parameters makes them difficult to use. In contrast, hybrid approaches achieve a better balance across multiple objectives but are more computationally intensive. Furthermore, the prevalence of scientific datasets (e.g., Montage, SIPHT, CyberShake) underscores a persistent gap in dataset realism, emphasizing the need for mixed or IoT-based benchmarks. The traditional EWA models remain static, optimizing only before execution, while the new taxonomy exposes the need for intelligent, self-adjusting systems that can react to changing cloud conditions. This study addresses the limits of earlier surveys and provides a basis for designing scalable, adaptive, and energy-aware workflow scheduling models for future cloud environments.

6 Discussion

In figure 2 percentage of the workflow applications that are used for implementing and validating of different workflow allocation approaches to optimize conflicting objectives. As shown in this figure, single workflow has been used more by authors. Figure 3 exhibits percentage of number workflows that used for implementing different workflow allocation approaches to optimize conflicting objectives. As shown in this figure only fewer schemes MWSTR[37], HPEFT[17], HEFT-ACO[29], PMWS-HC[61], SEPSO[62] have focused on multiple workflows and most author prefer to single workflow for allocation, and in this context multiple workflow allocation can be considered an active research area in the green cloud computing. In figure 4 exhibits percentage of workflow allocation objectives considered by different literature. As shown in this figure, most of the approaches studied multiobjective optimization, only fewer algorithms ASFLA[47], REEWS[14], HPEFT[17], EM_WOA[42] have employed single objective. Figure 6 shows percentage of QoS objectives used for optimization. Most of the studies used energy consumption (28%), makespan (27%), monetary cost (23%), and resource utilization (11%) for optimization criterion. These challenges pertain to EWAs, a feature that is highly prioritized by CSPs and frequently requested by users.

As shown in this figure, the objectives viz. load balancing (2%), response time, security (2%), throughput (3%), fault tolerance (2%) and convergence ration (1%) are considered in fewer studies. The researchers reported the percentage workflow allocation constraints in figure 7. As shown in this figure most the studied used deadline (46%) as optimization constraint, viz., EEWS[44],HAED[12],REEWS[14],CEAS [15],DRAWS [16],CPSO[19],MOGA[20],GA_PSO[26],ACRR[41],EM _WOA[42],EATS[48],COSA[58],EIS[60],SEPSO[62].

The figure 5 illustrates the distribution of workflow allocation approaches, including heuristic, metaheuristic, and hybrid methods, over the past nine years. The metaheuristic techniques have been the primary focus, accounting for (33%) of research in EWA challenges, while heuristic approaches have attracted (39%) of studies. In comparison, hybrid approaches have received minimal attention, with only (28%) of papers addressing the EWA problem HBMMO[25], GA-PSO[26], EEWS[44], HAED[12], HGALO-SCA[28], ANFIS[52], DCMORL[46], HCGWO[30], HEFT-ACO[29], ALPSO[31], HSMO[23], EBABC-PF[24],COSA [58].

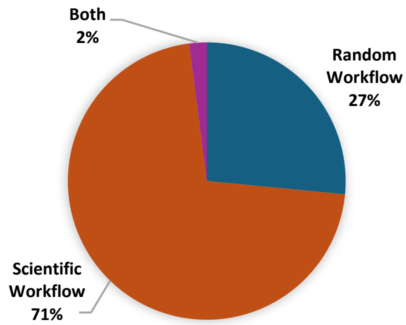

Figure 2: The frequency of workflow applications

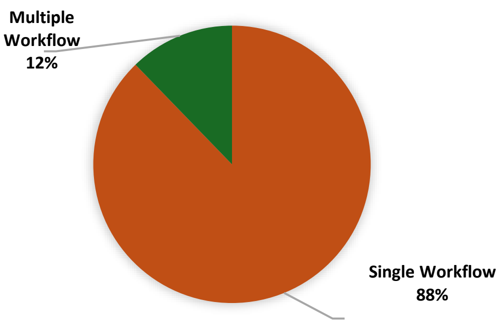

Figure 3: The frequency of workflows

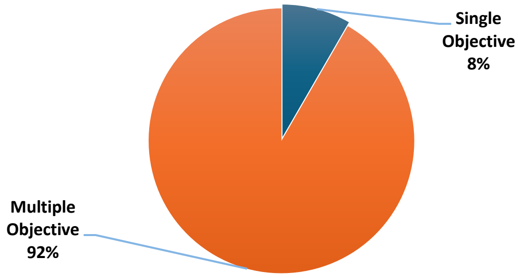

Figure 4: The frequency of Objectives

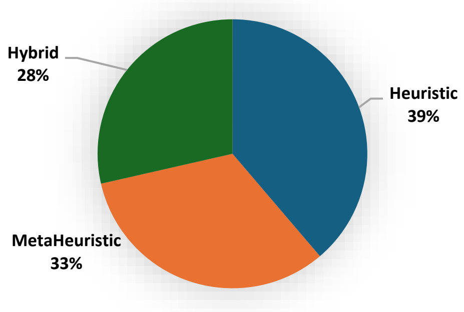

Figure 5: The frequency of WA approaches

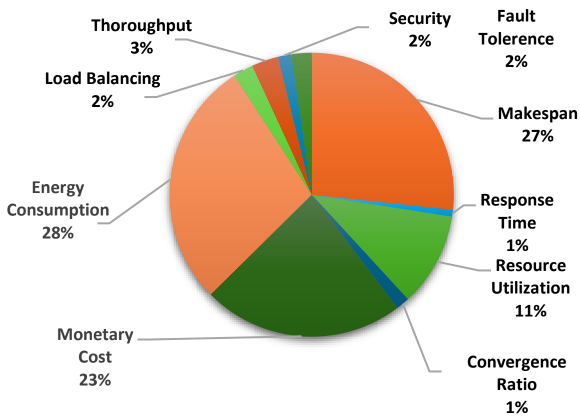

Figure 6: The frequency of QoS objectives

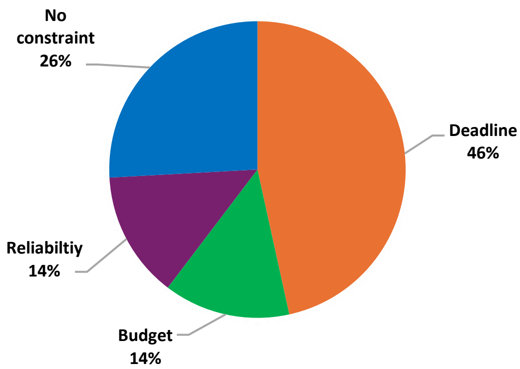

Figure 7: The frequency of QoS constraints

6.1 Analysis of EWA models

The comparative analysis of EWA models addresses RQ3 and reveals significant variation in performance depending on algorithmic design, workflow type, and environmental conditions as shown in Table 9. According to taxonomy, EWA techniques can be broadly grouped into heuristic, meta-heuristic, and hybrid approaches, each excelling under different operational constraints. These categorizations provide deeper insight into algorithm suitability across heterogeneous cloud environments.

The heuristic models are effective in static, small-scale environments where task dependencies and workload characteristics remain predictable. Their simplicity and low computational cost make them suitable for cost-driven or deadline-bound scenarios. However, they exhibit limited adaptability to dynamic workloads and fluctuating resource availability, leading to sub-optimal energy utilization in real-time or multi-workflow contexts.

In contrast, meta-heuristic models achieve a more effective balance between exploration and exploitation, which facilitates improved convergence toward Pareto-optimal solutions across multiple objectives. Their adaptability and diverse stochastic search strategies enhance resilience in dynamic and uncertain environments.

The hybrid models leverage both deterministic and probabilistic algorithm strengths to achieve the best trade-off between energy consumption, cost, and makespan. However, they also incur higher computational complexity, limiting their scalability for large, data-intensive workflows unless adaptive parameter tuning or reinforcement learning mechanisms are incorporated.

Table 9: Comparative analysis of algorithm

Algorithm Type	Representative Models	Key Observation
Heuristic	[13], [14], [15] , [16],[17], [35], [36], [37], [38], [39], [41], [43], [45], [48], [49], [50], [56], [57], [61]	Stable and fast but limited adaptability to dynamic workloads
Meta-Heuristic	[18] ,[19] [20] , [21] , [22] , [40] , [42] ,[47], [51], [53] ,[54], [55], [59], [60], [62], [63]	Good trade-offs; scalable but sensitive to parameters
Hybrid	[44] , [12] , [23] , [24] ,[25], [26], [27], [28], [29], [30] , [31] ,[46] ,[52], [58]	Best multi-objective performance; handles dynamic workflows efficiently

7 Open issues, challenges, and future trends

This section examines open issues, emerging trends, and key challenges associated with the EWA model in cloud computing systems.

7.1 Open issues

Here are some open issues related to energy-efficient workflow allocation in cloud environments, compiled based on recent research advancements. The following are a few important open issues:

7.1.1 Energy-aware resource allocation

Implementing energy-saving mechanisms, such as VM migration EASVMC[55] , CAAS[39] ,OWS-MRL[45], EnReal[36], RMREC[38], or dynamic voltage and frequency scaling (DVFS) MWSTR[37], EATS[48], BDCE, BDD[57], ACRR[41], COSA[58] can introduce additional overhead, which may offset the energy savings. Inefficient allocation strategies can lead to underutilized or overutilized resources, causing energy wastage or performance degradation. Many meta-heuristic algorithms suffer from slow convergence, particularly in high dimensional or complex problem spaces.

7.1.2 Monetary cost

The research indicates that the monetary cost for workflow execution in the IaaS cloud environment is a significant issue DRAWS[16], ASFLA[47], ECMSMOO[18], C-PSO[19], IWDC[22], MOGA [20], HBMMO[25], GA-PSO [26], HGAABC [27], HAED[12], HGALO-SCA[28], RMREC[38], OWS-MRL[45], ANFIS[52], I_MaOPSO[53], DCMORL[46], HCGWO[30], EAFSAIPR[54], HEFT-ACO[29], PACS[56], ACRR[41], HSMO[23], EBABC-PF [24], COSA[58], PMWS-HC[61], SEPSO[62]. The CSPs offer different configurations, capacities, and pricing structures for VMs. It is important to consider workflow requirements, VM characteristics, and associated pricing models when selecting an optimal service. Consequently, monetary cost represents an important factor in practical workflow processing.

7.1.3 Energy optimization in data-intensive workflows

In the studied literature data-intensive workflows are characterized by substantial data generation, processing, transfer, and storage requirements, making them common in fields such as big data analytics, scientific simulations and IoT applications IWDC[22],HBMMO[25],GA-PSO[26], EEWS[44], SERAS[35],ANFIS[52],DCMORL[46],HCGWO[30],EASVMC [55], HSMO[23], BDCE,BDD [57], EBABC-PF[24],EIS[60],PMWS-HC[61]. According to the existing literature, data-intensive workflows across domains such as big data analytics, scientific modelling, and Internet of Things (IoT) applications involve handling large volumes of data throughout their generation, processing, transfer, and storage. The energy consumption optimizing for these workflows is crucial, as their high resource demands can lead to significant power usage and operational costs.

7.1.4 Security aware resource allocation

Security remains a significant challenge in energy-efficient workflow allocation, particularly regarding data access, storage, and placement in cloud-based workflow systems EAFSAIPR[54], ALPSO[31], GA-PSO [26], ELSCiW[63]. The security remains a significant challenge in energy-efficient workflow allocation, particularly regarding data access, storage, and placement in cloud-based workflow systems. The risk of cyberattacks targeting critical data such as real-time workflow applications, financial transactions, e-commerce platforms, government computing systems, and healthcare electronics requires stronger security frameworks.

There are issues related to trust, data privacy, and secure data transmission that remain major obstacles in developing efficient and secure workflow allocation models, and addressing these challenges is important in achieving a balance between energy efficiency and robust security in cloud-based workflow scheduling.

7.1.5 Meta-Heuristic and hybrid optimization

According to the studied literature in cloud computing and workflow scheduling, meta-heuristic optimization techniques and hybrid optimization techniques have shown promise in resolving challenging optimization issues as shown in figure 5. However, challenges related to convergence speed, solution quality, and adaptability persist EATTO[40], C-PSO[19], EASVMC[55], HCGWO[30], CAAS[39]. The hybrid optimization techniques combine the strengths of multiple algorithms. These methods are highly adaptable to diverse problem domains and effectively address complex constraints and large-scale datasets.

7.2 Challenges

The EWA in cloud environments has emerged as a significant research focus, driven by the growing need for sustainable and cost-effective computing solutions. The literature identifies several key challenges in this domain:

7.2.1 Balancing energy efficiency and performance

The energy optimization measures often cause trade-offs in critical performance metrics, such as delayed execution time, increased latency, reduced throughput, or slower response times. This becomes even more critical in environments were user experience and SLA compliance, as shown in figure 6. The delayed task execution, missed deadlines, and unsatisfactory service quality can directly affect user satisfaction and trust in cloud services. The algorithms designed for energy-efficient workflow allocation must take a multi-objective optimization approach DRAWS[16] , ECMSMOO [18], HAED[12], HGALO-SCA [28], RMREC[38] ,OWS-MRL[45], ALPSO[31], PACS[56], ACRR [41].

7.2.2 Multi/Hybrid-Cloud environment optimization

Cloud computing has shifted from single cloud systems to multi-cloud and hybrid cloud setups. These newer models use both private and public resources or work with several providers, such as CEAS [15], EnReal[36] , HEFT-ACO[29],PMWS-HC[61], ELSCiW [63]. Although these environments offer cost savings, scalability, and flexibility, it is still challenging to optimize process allocation in these diverse and changing environments.

7.2.3 Dynamic environment

Cloud environments are always changing, with workloads, resource needs, and energy use shifting all the time. Data-intensive workflows can also have unpredictable data volumes and processing needs, which makes static energy optimization less effective. In addition, VM migration and DVFS techniques can add operational overhead that limits efficiency gains, as shown in studies like EASVMC[55] , CAAS[39] ,OWS-MRL[45], EnReal[36], RMREC[38].

7.2.4 Scientific workflow application

Scientific workflow applications have complex dependencies and need a lot of resources, which makes it hard to allocate energy efficiently in cloud computing as shown in figure 2 and figure 3. Many such workflows involve heavy data processing and transfer across distributed cloud resources, which can significantly increase energy consumption. Scientific workflows often have fluctuating resource requirements, making it difficult to predict and allocate resources efficiently. Scientific workflows require high reliability to ensure the accuracy of results DRAWS[16] ,SERAS [35], EERS [13], ACRR [41]. Moreover, it must meet stringent QoS parameters, including reliability, security, and execution deadlines, while optimizing energy efficiency. Scientific workflows often have fluctuating resource requirements, making it difficult to predict and allocate resources efficiently.

7.2.5 Multi-Objective optimization

The reviewed literature makes it clear that while some QoS constraints of EWA for systems were considered, others were not. In certain models, for instance, deadline and budget (MOGA [20], GA-PSO [26], HSMO [23], BDCE, BDD [57], COSA[58] , SEPSO[62]), as well as deadline and reliability DRAWS[16] , REEWS[14], SERAS [35], I_MaOPSO[53]), are carefully considered, whereas other QoS constraints are disregarded as shown in figure 7. Therefore, the optimal method that considers many objectives to balance different QoS factors in EWA for cloud computing may still be challenging.

7.3 Future trends

This section presents potential future opportunities and research directions for EWA models within computing environments. Future trends are categorized according to the literature, with relevant factors including workflow-as-a-service, many-objective optimization, fog and edge computing, and green computing.

7.3.1 Workflow-as-a-Service

Workflow-as-a-Service (WaaS) has gained attention as a scalable approach to managing multiple workflow instances under dynamic workloads[1]. WaaS manages multiple workflow requests with varying arrival patterns, requiring efficient task scheduling and resource management. These workflows, often represented as DAGs and the services must be allocated to cloud resources dynamically. The system dynamically adjusts resource allocation in response to workload fluctuations, incorporating cost-effective virtual machine leasing strategies. Although WaaS has been increasingly adopted in areas such as online batch processing, supply chain management, and e-commerce, current research demonstrates limited advancement in EWA models for multiple workflow applications within cloud environments. Furthermore, only a small number of models have been proposed for workflow scheduling in cloud computing.

7.3.2 Many objective optimization

In cloud computing, optimizing many conflicting objectives such as energy efficiency, security, and budget constraints while ensuring workflow execution within given constraints remains a critical challenge[53]. To achieve an optimal trade-off among these objectives, it is essential to determine the values of all decision variables effectively. Many objectives optimization, particularly in the context of evolutionary algorithms, relies on Pareto dominance to evaluate potential solutions. However, as the number of objectives increases, maintaining a balance between convergence and diversity becomes increasingly difficult CAAS[39], ANFIS[52] , EAFSAIPR[54], ALPSO[31] , ACRR[41].

7.3.3 Fog/Edge computing

Fog computing moves cloud capabilities closer to where data is created by allowing processing at the edge of the network. This setup means less data needs to travel to central data centers, which lowers both delays and energy use. It also reduces the load on main servers and makes the whole system more energy efficient. Edge computing works in a similar way, handling data processing and storage near where the data is generated. These methods help systems respond faster and save significant energy.

7.3.4 Green computing

As adoption of cloud services growing quickly, adopting Green Cloud Computing (GCC) has become important for long-term sustainability. The data centers use a lot of electricity and are major sources of global carbon emissions. It focuses on making cloud systems more energy efficient and environmentally responsible by improving infrastructure, cutting power use, and using renewable energy. The sustainability strategies include efficient scheduling, virtualization, and resource optimization to lower the carbon footprint. Because of these challenges, researchers working on cloud scheduling algorithms need to find solutions that support greener computing.

7.3.5 AI & ML-driven energy optimization

Cloud computing is moving toward using Artificial Intelligence (AI) and Machine Learning (ML) to boost energy efficiency, make better use of resources, and support sustainability. With AI and ML, workflow scheduling gets smarter, allowing for more flexible resource use and saving energy. As a result, the right resources are available at the right time, which helps manage energy use.

7.3.6 Integrating Adaptive Control concepts into EWA

Recent developments in adaptive control theory, such as adaptive fuzzy control, neural adaptive control, and backstepping control, are helping to improve EWA in cloud systems. Fuzzy control uses logical rules to estimate the relationship between workload and energy use, allowing self-adjusting resource management. Neural adaptive control uses learning techniques to dynamically modify task scheduling based on feedback such as VM load and energy data. When these adaptive methods are combined with meta-heuristic optimization and feedback learning, EWA systems can automatically balance energy efficiency, performance, and reliability as conditions change.

8 Conclusion

Energy-efficient workflow allocation (EWA) in cloud computing plays a key role in optimizing resource utilization and minimizing power consumption. Although there has been a lot of progress, persistent challenges remain, including load balancing, performance optimization, security, economic cost, and energy efficiency in cloud environments. This paper surveys recent work on energy-efficient workflow scheduling, reviews EWA challenges, and discusses open issues and future research. The study begins by analyzing the fundamental components of workflow allocation, including workflow models, key definitions, deployment environments, application domains, workflow approaches, workload classifications, and QoS objectives and constraints. We first surveyed recent related survey papers for the EWA to find research gaps. Secondly, we discuss workflow allocation components, including the deployment model, workflow application and models, EWA approach, workload types, QoS objectives, and constraints, followed by a general problem statement of EWA. A taxonomy of EWA challenges in cloud environments is presented, along with an allocation framework for the EWA model to enhance understanding of the problem. Finally, the paper discusses open issues, challenges, and future research directions to support the development of advanced EWA approaches. Finally, the various open issues, challenges, and future directions for further research are discussed to help the researchers develop EWA approaches in the domain. The analysis of energy-efficient workflow allocation indicates that energy consumption (28%), makespan (27%), and monetary cost (23%) are the primary concerns in current research. In contrast, aspects such as fault tolerance, response time, and convergence ratio receive considerably less attention. As current studies focus most on QoS factors like task deadlines, which make up 46% of the research. In contrast, budget and reliability are each considered in only about 14% of studies. Future research should focus on scalable, communication-aware, and adaptive multiobjective optimization methods that can improve energy efficiency, performance, security, and cost at the same time. Using hybrid meta-heuristic strategies will also be important for more effective and balanced workflow scheduling. This study aims to provide valuable insights into energy-efficient workflow allocation, guiding future advancements in sustainable, high-performance cloud computing infrastructure.

References

[1] F. Wu, Q. Wu, and Y. Tan, ‘Workflow scheduling in cloud: a survey’, Journal of Supercomputing, vol. 71, no. 9, pp. 3373–3418, Sep. 2015.

https://doi.org/10.1007/s11227-015-1438-4

[2] M. Masdari, S. ValiKardan, Z. Shahi, and S. I. Azar, ‘Towards workflow scheduling in cloud computing: A comprehensive analysis’, Journal of Network and Computer Applications, vol. 66, pp. 64–82, 2016.

https://doi.org/10.1016/j.jnca.2016.01.018

[3] M. A. Rodriguez and R. Buyya, ‘A taxonomy and survey on scheduling algorithms for scientific workflows in IaaS cloud computing environments’, Concurrency Computation, vol. 29, no. 8, Apr. 2017.

https://doi.org/10.1002/cpe.4041

[4] M. Adhikari, T. Amgoth, and S. N. Srirama, ‘A survey on scheduling strategies for workflows in cloud environment and emerging trends’, ACM Computing Surveys, vol. 52, no. 4, Aug. 2019.

https://doi.org/10.1145/3325097

[5] Hosseinzadeh, M., Ghafour, M.Y., Hama, H.K. et al. Multi-Objective Task and Workflow Scheduling Approaches in Cloud Computing: a Comprehensive Review. J Grid Computing 18, 327–356 (2020).

https://doi.org/10.1007/s10723-020-09533-z

[6] L. Versluis and A. Iosup, ‘A survey of domains in workflow scheduling in computing infrastructures: Community and keyword analysis, emerging trends, and taxonomies’, Future Generation Computer Systems, vol. 123, pp. 156–177, 2021.

https://doi.org/10.1016/j.future.2021.04.009

[7] Y. Kumar, S. Kaul, and Y.-C. Hu, ‘Machine learning for energy-resource allocation, workflow scheduling and live migration in cloud computing: State-of-the-art survey’, Sustainable Computing: Informatics and Systems, vol. 36, p. 100780, Dec. 2022.

https://doi.org/10.1016/j.suscom.2022.100780

[8] R. Medara and R. S. Singh, A Review on Energy-Aware Scheduling Techniques for Workflows in IaaS Clouds, vol. 125, no. 2. Springer US, 2022.

https://doi.org/10.1007/s11277-022-09621-1

[9] P. Verma, A. K. Maurya, and R. S. Yadav, ‘A survey on energy-efficient workflow scheduling algorithms in cloud computing’, Software Practice and Experience, vol. 54, no. 5, pp. 637–682, May 2024.

https://doi.org/10.1002/spe.3292

[10] M. M. Nezami, A. Kumar, M. Shahid, and M. M. Nezami, ‘Analysis of Energy Efficient Workflow Allocation in Cloud Computing’, in 2023 IEEE 12th International Conference on Communication Systems and Network Technologies (CSNT), , Apr. 2023, pp.803 -809.

https://doi.org/10.1109/csnt57126.2023.10134644

[11] S. Suresh and S. Sakthivel, ‘A novel performance constrained power management framework for cloud computing using an adaptive node scaling approach’, Computers and Electrical Engineering, vol. 60, pp. 30–44, May 2017.

https://doi.org/10.1016/j.compeleceng.2017.04.018

[12] M. Kalra and S. Singh, ‘Multi-objective Energy Aware Scheduling of Deadline Constrained Workflows in Clouds using Hybrid Approach’, Wireless Personal Communications, vol. 116, no. 3, pp. 1743–1764, 2021.

https://doi.org/10.1007/s11277-020-07759-4

[13] R. Medara and R. S. Singh, ‘Energy Efficient and Reliability Aware Workflow Task Scheduling in Cloud Environment’, Wireless Personal Communications, vol. 119, no. 2, pp. 1301–1320, Jul. 2021.

https://doi.org/10.1007/s11277-021-08263-z

[14] R. Garg, M. Mittal, and L. H. Son, 'Reliability and energy efficient workflow scheduling in cloud environment', Cluster Computing, vol. 22, no. 4, pp. 1283–1297, Dec. 2019.

https://doi.org/10.1007/s10586-019-02911-7

[15] Z. Li, J. Ge, H. Hu, W. Song, H. Hu, and B. Luo, 'Cost and Energy Aware Scheduling Algorithm for Scientific Workflows with Deadline Constraint in Clouds', IEEE Transactions on Services Computing, vol. 11, no. 4, pp. 713–726, Jul. 2018.

https://doi.org/10.1109/tsc.2015.2466545

[16] H. Ji, W. Bao, and X. Zhu, 'Adaptive workflow scheduling for diverse objectives in cloud environments', Transactions on Emerging Telecommunications Technologies, vol. 28, no. 2, Feb. 2017.

https://doi.org/10.1002/ett.2941

[17] Y. Li, W. Tang, and G. Liu, 'HPEFT for hierarchical heterogeneous multi-DAG in a multigroup scan UPA system', Electronics (Switzerland), vol. 8, no. 5, May 2019.

https://doi.org/10.3390/electronics8050498

[18] G. Yao, Y. Ding, Y. Jin, and K. Hao, 'Endocrine-based coevolutionary multi-swarm for multi-objective workflow scheduling in a cloud system', Soft Computing - A Fusion of Foundations, Methodologies and Applications, vol. 21, no. 15, pp. 4309–4322, Aug. 2017.

https://doi.org/10.1007/s00500-016-2063-8

[19] S. J. Nirmala and S. M. S. Bhanu, 'Catfish-PSO based scheduling of scientific workflows in IaaS cloud', Computing, vol. 98, no. 11, pp. 1091–1109, 2016.

https://doi.org/10.1007/s00607-016-0494-9

[20] A. Rehman, S. S. Hussain, Z. ur Rehman, S. Zia, and S. Shamshirband, 'Multi-objective approach of energy efficient workflow scheduling in cloud environments', Concurrency Computing, vol. 31, no. 8, pp. 1–20, 2019.

https://doi.org/10.1002/cpe.4949

[21] N. Rehani and R. Garg, 'Meta-heuristic based reliable and green workflow scheduling in cloud computing', International Journal of System Assurance Engineering and Management, vol. 9, no. 4, pp. 811–820, Aug. 2018.

https://doi.org/10.1007/s13198-017-0659-8

[22] S. Elsherbiny, E. Eldaydamony, M. Alrahmawy, and A. E. Reyad, 'An extended Intelligent Water Drops algorithm for workflow scheduling in cloud computing environment', Egyptian Informatics Journal, vol. 19, no. 1, pp. 33–55, Mar. 2018.

https://doi.org/10.1016/j.eij.2017.07.001

[23] N. Rizvi, R. Dharavath, and D. R. Edla, 'Cost and makespan aware workflow scheduling in IaaS clouds using hybrid spider monkey optimization', Simulation Modelling Practice and Theory, vol. 110, no. October 2020, p. 102328, 2021.

https://doi.org/10.1016/j.simpat.2021.102328

[24] M. Zeedan, G. Attiya, and N. El-Fishawy, 'Enhanced hybrid multi-objective workflow scheduling approach based artificial bee colony in cloud computing', Computing, 2022.

https://doi.org/10.1007/s00607-022-01116-y

[25] N. Anwar and H. Deng, 'A hybrid metaheuristic for multi-objective scientific workflow scheduling in a cloud environment', Applied Sciences (Switzerland), vol. 8, no. 4, Mar. 2018.

https://doi.org/10.3390/app8040538

[26] A. M. Manasrah and H. B. Ali, 'Workflow Scheduling Using Hybrid GA-PSO Algorithm in Cloud Computing', Wireless Communications and Mobile Computing, vol. 2018.

https://doi.org/10.1155/2018/1934784

[27] Y. Gao, S. Zhang, and J. Zhou, 'A Hybrid Algorithm for Multi-Objective Scientific Workflow Scheduling in IaaS Cloud', IEEE Access, vol. 7, pp. 125783–125795, 2019.

https://doi.org/10.1109/access.2019.2939294

[28] A. Mohammadzadeh, M. Masdari, F. S. Gharehchopogh, and A. Jafarian, 'A hybrid multi-objective metaheuristic optimization algorithm for scientific workflow scheduling', Cluster Computing, vol. 24, no. 2, pp. 1479–1503, Jun. 2021.

https://doi.org/10.1007/s10586-020-03205-z

[29] A. Belgacem and K. Beghdad-Bey, 'Multi-objective workflow scheduling in cloud computing: trade-off between makespan and cost', Cluster Computing, vol. 25, no. 1, pp. 579–595, Feb. 2022.

https://doi.org/10.1007/s10586-021-03432-y

[30] A. Mohammadzadeh, M. Masdari, F. S. Gharehchopogh, and A. Jafarian, 'Improved chaotic binary grey wolf optimization algorithm for workflow scheduling in green cloud computing', Evolutionary Intelligence, vol. 14, no. 4, pp. 1997–2025, Dec. 2021.

https://doi.org/10.1007/s12065-020-00479-5

[31] J. Kakkottakath Valappil Thekkepuryil, D. P. Suseelan, and P. M. Keerikkattil, 'An effective meta-heuristic based multi-objective hybrid optimization method for workflow scheduling in cloud computing environment', Cluster Computing, vol. 24, no. 3, pp. 2367–2384, 2021.

https://doi.org/10.1007/s10586-021-03269-5

[32] M. M. Nezami, A. Kumar, 'Energy Efficient Workflow Allocation in Cloud Computing Using Improved Grey Wolf Optimization', International Journal of Advanced Computer Science and Applications, vol. 16, no. 10, pp. 491–498, Oct. 2025.

https://doi.org/10.14569/ijacsa.2025.0161051

[33] M. Shahid, Z. Raza, and M. Sajid, 'Level based batch scheduling strategy with idle slot reduction under DAG constraints for computational grid', Journal of Systems and Software, vol. 108, pp. 110–133, 2015.

https://doi.org/10.1016/j.jss.2015.06.016

[34] W. Zheng and S. Huang, 'Deadline Constrained Energy-Efficient Scheduling for Workflows in Clouds', Second International Conference on Advanced Cloud and Big Data, IEEE, pp.69–76, Nov. 2014.

https://doi.org/10.1109/cbd.2014.18

[35] H. A. Hassan, S. A. Salem, and E. M. Saad, 'A smart energy and reliability aware scheduling algorithm for workflow execution in DVFS-enabled cloud environment', Future Generation Computer Systems, vol. 112, pp. 431–448, Nov. 2020.

https://doi.org/10.1016/j.future.2020.05.040

[36] X. Xu, W. Dou, X. Zhang, and J. Chen, 'EnReal: An Energy-Aware Resource Allocation Method for Scientific Workflow Executions in Cloud Environment', IEEE Transactions on Cloud Computing, vol. 4, no. 2, pp. 166–179, Apr. 2016.

https://doi.org/10.1109/tcc.2015.2453966

[37] J. Jiang, Y. Lin, G. Xie, L. Fu, and J. Yang, 'Time and Energy Optimization Algorithms for the Static Scheduling of Multiple Workflows in Heterogeneous Computing System', J Grid Computing, vol. 15, no. 4, pp. 435–456, Dec. 2017.

https://doi.org/10.1007/s10723-017-9391-5

[38] L. Zhang, L. Wang, Z. Wen, M. Xiao, and J. Man, 'Minimizing Energy Consumption Scheduling Algorithm of Workflows with Cost Budget Constraint on Heterogeneous Cloud Computing Systems', IEEE Access, vol. 8, pp. 205099–205110, 2020.

https://doi.org/10.1109/access.2020.3037205

[39] R. Ranjan, I. S. Thakur, G. S. Aujla, N. Kumar, and A. Y. Zomaya, 'Energy-Efficient Workflow Scheduling Using Container-Based Virtualization in Software-Defined Data Centers', IEEE Transactions on Industrial Informatics, vol. 16, no. 12, pp. 7646–7657, Dec. 2020.

https://doi.org/10.1109/tii.2020.2985030

[40] Y. Gu and C. Budati, 'Energy-aware workflow scheduling and optimization in clouds using bat algorithm', Future Generation Computer Systems, vol. 113, pp. 106–112, Dec. 2020.

https://doi.org/10.1016/j.future.2020.06.031

[41] S. Danthuluri and S. Chitnis, 'Energy and cost optimization mechanism for workflow scheduling in the cloud', Materials Today: Proceedings, Jul. 2021.

https://doi.org/10.1016/j.matpr.2021.07.168

[42] L. Zhang, L. Wang, M. Xiao, Z. Wen, and C. Peng, 'EM_WOA: A budget-constrained energy consumption optimization approach for workflow scheduling in clouds', Peer-to-Peer Netw. Appl., vol. 15, no. 2, pp. 973–987, 2022.

https://doi.org/10.1007/s12083-021-01267-3

[43] Neha Garg, Neeraj, Manish Raj, Indrajeet Gupta, Vinay Kumar and G. R. Sinha, 'Energy-aware scientific workflow scheduling in cloud environment', Cluster Computing, vol. 2022, 2022.

https://doi.org/10.1155/2022/1637614

[44] V. Singh, I. Gupta, and P. K. Jana, 'An Energy Efficient Algorithm for Workflow Scheduling in IaaS Cloud', J Grid Computing, vol. 18, no. 3, pp. 357–376, Sep. 2020.

https://doi.org/10.1007/s10723-019-09490-2

[45] A. Asghari, M. K. Sohrabi, and F. Yaghmaee, 'Online scheduling of dependent tasks of cloud's workflows to enhance resource utilization and reduce the makespan using multiple reinforcement learning-based agents', Soft computing, vol. 24, no. 21, pp. 16177–16199, Nov. 2020.

https://doi.org/10.1007/s00500-020-04931-7

[46] Y. Qin, H. Wang, S. Yi, X. Li, and L. Zhai, 'A multi-objective reinforcement learning algorithm for deadline constrained scientific workflow scheduling in clouds', Frontiers of Computer Science, vol. 15, no. 5, p. 155105, Oct. 2021.

https://doi.org/10.1007/s11704-020-9273-z

[47] P. Kaur and S. Mehta, 'Resource provisioning and workflow scheduling in clouds using augmented Shuffled Frog Leaping Algorithm', Journal of Parallel and Distributed Computing, vol. 101, pp. 41–50, Mar. 2017.

https://doi.org/10.1016/j.jpdc.2016.11.003

[48] M. Safari and R. Khorsand, 'Energy-aware scheduling algorithm for time-constrained workflow tasks in DVFS-enabled cloud environment', Simulation Modelling Practice and Theory, vol. 87, pp. 311–326, Sep. 2018.

https://doi.org/10.1016/j.simpat.2018.07.006

[49] X. Qu, P. Xiao, and L. Huang, 'Improving the energy efficiency and performance of data-intensive workflows in virtualized clouds', Journal of Supercomputing, vol. 74, no. 7, pp. 2935–2955, Jul. 2018.

https://doi.org/10.1007/s11227-018-2344-3

[50] B. Qureshi, 'Profile-based power-aware workflow scheduling framework for energy-efficient data centers', Future Generation Computer Systems, vol. 94, pp. 453–467, May 2019.

https://doi.org/10.1016/j.future.2018.11.010

[51] S. Gupta, I. Agarwal, and R. S. Singh, 'Workflow scheduling using Jaya algorithm in cloud', Concurrent Computing, vol. 31, no. 17, 2019.

https://doi.org/10.1002/cpe.5251

[52] M. Alaei, R. Khorsand, and M. Ramezanpour, 'An adaptive fault detector strategy for scientific workflow scheduling based on improved differential evolution algorithm in cloud', Applied Soft Computing, vol. 99, p. 106895, Feb. 2021.

https://doi.org/10.1016/j.asoc.2020.106895

[53] S. Saeedi, R. Khorsand, S. Ghandi Bidgoli, and M. Ramezanpour, 'Improved many-objective particle swarm optimization algorithm for scientific workflow scheduling in cloud computing', Computers & Industrial Engineering, vol. 147, p. 106649, Sep. 2020.

https://doi.org/10.1016/j.cie.2020.106649

[54] A. Ramathilagam and K. Vijayalakshmi, 'Workflow scheduling in cloud environment using a novel metaheuristic optimization algorithm', International Journal of Communication Systems, vol. 34, no. 5, Mar. 2021.

https://doi.org/10.1002/dac.4746

[55] R. Medara, R. S. Singh, and Amit, 'Energy-aware workflow task scheduling in clouds with virtual machine consolidation using discrete water wave optimization', Simulation Modelling Practice and Theory, vol. 110, no. March, p. 102323, 2021.

https://doi.org/10.1016/j.simpat.2021.102323

[56] M. A. Khan, 'A cost-effective power-aware approach for scheduling cloudlets in cloud computing environments', Journal of Supercomputing, vol. 78, no. 1, pp. 471–496, Jan. 2022.

https://doi.org/10.1007/s11227-021-03894-2

[57] A. Taghinezhad-Niar, S. Pashazadeh, and J. Taheri, 'Energy-efficient workflow scheduling with budget-deadline constraints for cloud', Computing, vol. 104, no. 3, pp. 601–625, Mar. 2022.

https://doi.org/10.1007/s00607-021-01030-9

[58] H. Li, G. Xu, D. Wang, M. Zhou, Y. Yuan, and A. Alabdulwahab, 'Chaotic-Nondominated-Sorting Owl Search Algorithm for Energy-Aware Multi-Workflow Scheduling in Hybrid Clouds', IEEE Transactions on Sustainable Computing, vol. 7, no. 3, pp. 595–608, Jul. 2022.

https://doi.org/10.1109/tsusc.2022.3144357

[59] A. Choudhary, M. C. Govil, G. Singh, L. K. Awasthi, and E. S. Pilli, 'Energy-aware scientific workflow scheduling in cloud environment', Cluster Computing, 2022.

https://doi.org/10.1007/s10586-022-03613-3

[60] M. Cao, Y. Li, X. Wen, Y. Zhao, and J. Zhu, 'Energy-aware intelligent scheduling for deadline-constrained workflows in sustainable cloud computing', Egyptian Informatics Journal, vol. 24, no. 2, pp. 277–290, Jul. 2023.

https://doi.org/10.1016/j.eij.2023.04.002

[61] Z. Sun, H. Huang, Z. Li, C. Gu, R. Xie, and B. Qian, 'Efficient, economical and energy-saving multi-workflow scheduling in hybrid cloud', Expert Systems with Applications, vol. 228, p. 120401, Oct. 2023.

https://doi.org/10.1016/j.eswa.2023.120401

[62] H. Li, X. Li, J. Xu, and L. Chen, 'Entropy based swarm intelligent searching for scheduling deadline constrained workflows in hybrid cloud', International Journal of Machine Learning and Cybernetics, vol. 15, no. 4, pp. 1183–1199, Apr. 2024.

https://doi.org/10.1007/s13042-023-01962-y

[63] M. I. Khaleel, M. Safran, S. Alfarhood, and M. Zhu, 'Energy-latency trade-off analysis for scientific workflow in cloud environments: The role of processor utilization ratio and mean grey wolf optimizer', Engineering Science and Technology, an International Journal, vol. 50, p. 101611, Feb. 2024.

https://doi.org/10.1016/j.jestch.2023.101611

Abbreviations

The list of abbreviations used in this study is given in alphabetical order below.

Abbreviation	Full form	Abbreviation	Full form
ABC	Artificial Bee Colony	GWO	Grey Wolf Optimization
ACO	Ant Colony Optimization	HAED	Hybrid Approach for Energy-Aware Scheduling of Deadline-Constrained Workflows
ACRR	Adaptive Cloud Resource Reconfigurability	HEFT	Heterogenous Earliest Finish Time
ALO	Ant Loin Optimization	HPEFT	Heterogeneous Predecessor Earliest Finish Time
ASFLA	Shuffled Frog Leaping Algorithm	IDE	Improved Differential Evolution
BDCE	Budget Deadline Constrained Energy-aware	IWD	Intelligent Water Drops
BDD	Budget Deadline DVFS-enabled Energy-aware	MCP	Modified Critical Path
BDSD	Budget Deadline Sensitive Dynamic	MOGA	Multiobjective Genetic Algorithm
CEAS	Cost and Energy Aware Scheduling	MSIA	Multiobjective Salp Swarm Algorithm
C-PSO	Catfish Particle Swarm Optimization	MWSTR	Multiple Workflow Slack Time Reclamation
CPFD	Critical Path Fast Duplication	NSGA-II	Nondominated Sorting Genetic Algorithm II
CPM	Critical Path Method	PACS	Power-Aware Cloudlet Scheduling
CSO	Cat Swarm Optimization	PSO	Particle Swarm Optimization
DES	Data Encryption Standard	QoS	Quality of Service
DVFS	Dynamic Voltage Frequency Scaling	REEWS	Reliability and Energy Efficient Workflow Scheduling
EAFSA	Enhanced Artificial Fish Swarm	SERAS	Smart Energy and Reliability Aware Scheduling
EASVMC	Energy Aware Workflow Scheduling System for Cloud Computing with VM Consolidation	SFLA	Shuffled Frog-Leaping Algorithm
ECMSMOO	Endocrine-based Coevolutionary Multi-Swarm for Multi-Objective Optimization	SLA	Service Level Agreement
EERS	Energy-Efficient and Reliability Aware Workflow Task Scheduling	SPEA2	Strength Pareto Evolutionary Algorithm 2
EEWS	Energy Efficient Workflow Scheduling	TDS	Task Duplication-based Scheduling
EnReal	Energy-aware Resource Allocation	WWO	Water Wave Optimization
EVMP	Energy-aware Virtual Machine Placement	WOA	Whale Optimization Algorithm