A PRISMA-Based Systematic Review of Cloud-Edge Orchestration Using the MAPE-K Framework

Nisha Saini* and Jitender Kumar
Department of Computer Science and Engineering, Deenbandhu Chhotu Ram University of Science and Technology, Murthal (Sonipat), India
Email: saini.nisha0203@gmail.com (N.S.), jitenderkumar.cse@dcrustm.org (J.K.)

Manuscript received February 5, 2025; revised March 23, 2025; accepted April 5, 2025
*Corresponding author

Abstract—**Although cloud computing offers abundant computational capacity, it suffers from inherent latency. Edge computing mitigates this by processing data closer to the edge of the infrastructure, thereby reducing latency and improving performance. However, challenges arise owing to inadequate edge processing capacity and significant cloud latency. A viable solution to address these challenges is cloud-edge integration, a collaborative resource distribution model. This study examines cloud-edge orchestration, focusing on performance and efficiency improvements through orchestration techniques. We performed a comprehensive systematic literature review using the PRISMA model, which compiled 10,389 records from the decade spanning 2015 to 2024, filtered down to 89 studies. Using the Monitor-Analyze-Plan-Execute over shared Knowledge (MAPE-K) framework, we assessed cloud-edge orchestration and categorized the performance metrics. Additionally, we evaluate the task distribution criteria between cloud and edge computing environments, identify challenges, and outline prospective directions. Our findings provide insights into optimizing cloud-edge computing for mainstream applications, ensuring improved resource management and efficiency in cloud-edge computing.**

Index Terms—**cloud-edge orchestration, MAPE-K framework, performance optimization, PRISMA model, resource provisioning, task distribution**

I. Introduction

Cloud computing is a highly beneficial technology that provides various software, platforms, and infrastructure services to users. It offers scalable and on-demand access to computing power, storage, and applications over the Internet. The benefits of cloud computing include cost-effectiveness, flexibility, and improved collaboration. However, it also faces challenges, such as data security concerns, potential downtime, and latency issues in certain applications. For instance, the demand for web applications is often highest during peak hours, with demand surging during midday and decreasing in the morning and evening hours. This variability often requires resource deployment management to ensure that consumers receive prompt responses. Edge computing has emerged as a complementary technology to address the limitations of cloud computing. It brings computation and data storage closer to the devices where they are needed, thereby reducing latency and bandwidth usage. Edge computing is crucial for Internet of Things (IoT) applications that require real-time processing and low latencies [1]. The need for edge computing arises from the increasing number of connected devices and the growing demand for faster response times. However, edge computing also faces challenges, including limited computational resources, security vulnerabilities, and complex management of distributed systems. These constraints inherent in IoT environments, such as the limited processing power, storage capacity, and energy resources of edge devices, coupled with the need for real-time data processing and low latency, necessitate the development and implementation of efficient orchestration techniques to optimize resource utilization and ensure seamless operation across the cloud-edge continuum.

In the context of cloud and edge computing, orchestration refers to the automated arrangement, coordination, and management of complex computer systems, middleware, and services. Cloud-edge orchestration involves managing the distribution of workloads and resources between the cloud and edge environments. This orchestration is necessary to optimize performance, ensure efficient resource utilization, and maintain service quality across the entire computing infrastructure. The need for cloud-edge orchestration arises from the complexity of managing hybrid environments that combine centralized cloud resources with distributed edge nodes. Effective orchestration can provide several benefits, including improved application performance, reduced network congestion, enhanced data privacy and security, and more efficient computing resource utilization. This allows organizations to leverage the strengths of both cloud and edge computing while mitigating their limitations.

Various techniques have been employed for cloud-edge orchestration to achieve these benefits. These include workload distribution algorithms that determine where to process data based on factors such as latency requirements, resource availability, and energy efficiency. Resource provisioning techniques ensure that adequate

doi: 10.18178/ijeetc.14.3.130-146

computing power, storage, and network resources are allocated to meet application demands. Task scheduling algorithms optimize the execution of tasks across cloud and edge resources to minimize response times and maximize resource utilization. Additionally, orchestration techniques often incorporate dynamic load balancing to distribute workloads evenly across available resources, preventing bottlenecks and ensuring optimal performance. Virtual Machine (VM) migration strategies are used to move computational tasks between cloud and edge environments as needed, adapting to changing conditions and requirements. Server consolidation techniques help optimize resource usage by efficiently grouping workloads into fewer physical servers. Advanced orchestration systems may also employ machine learning and artificial intelligence to predict resource requirements, detect anomalies, and automatically adjust resource allocations. These intelligent orchestration techniques can adapt to changing workloads and network conditions, ensuring consistent performance and efficient resource utilization across the cloud-edge continuum.

Cloud-edge orchestration faces several challenges, including heterogeneous resource management, dynamic workload allocation, network variability, and security concerns in distributed environments. To address these challenges, cloud-edge orchestration techniques often employ the Monitor-Analyze-Plan-Execute over shared Knowledge (MAPE-K) framework. This approach provides a structured method for managing complex systems. In the monitoring phase, data are collected from various cloud and edge resources, including the performance metrics, resource utilization, and network conditions. The analysis phase processes the data to identify patterns, anomalies, and potential issues. Based on this analysis, the planning phase determines the optimal resource allocation and workload distribution strategies. The execution phase implements these plans by adjusting resource allocation, migrating workloads, and reconfiguring network paths as needed. The shared knowledge component acts as a central repository of information, policies, and historical data that informs decision making across all phases. This framework enables continuous adaptation to changing conditions, ensuring efficient resource utilization and maintaining quality of service across the cloud-edge continuum. By implementing MAPE-K, organizations can create resilient, responsive, and efficient cloud-edge orchestration systems that can manage the complexities of modern distributed computing environments. Careful metric selection serves as a motivation for the evaluation process and helps to determine whether the technique achieves its objectives. Therefore, in Table I, we have endeavored to address the following Research Questions (RQs) and their purpose.

TABLE I: MOTIVATION FOR RESEARCH QUESTIONS

No.	RQs	Motivation
RQ1	To what extent does the MAPE-K approach effectively assess the performance of collaborative cloud-edge techniques?	To determine whether collaborative cloud-edge technologies can be effectively assessed using this approach.
RQ2	How can the performance of cloud-edge orchestration be measured?	To locate and scrutinize the performance measures pertinent to the orchestration of resources.
RQ3	What mechanisms determine the optimal distribution of tasks between cloud and edge nodes?	To optimize performance and efficiency, the distribution of tasks between edge devices and cloud resources must be understood.
RQ4	What are the open issues and preliminary challenges of cloud-edge orchestration?	A clear view of the inherent unresolved issues in orchestration is needed so that adopters can prepare for them accordingly.
RQ5	What are the opportunities and future trends in cloud-edge orchestration?	This study aims to understand which areas may be interesting to explore in the field of cloud-edge orchestration.

Several studies have meticulously investigated different aspects of edge, fog, and cloud computing [2]. These studies focused on different aspects, such as computation offloading, task scheduling, and optimally provisioned resources, and used different methods to solve these issues. These studies illuminated various strategies for optimizing computation offloading, task scheduling, and resource provisioning in edge-cloud contexts. For instance, the exploration of operational collaboration issues [3, 4], and a comparative analysis of different system architectures [5] have enriched our understanding of the complexities involved. Similarly, the focus on application partitioning techniques [6], critical edge computing applications [7], and innovative task offloading algorithms [8] has provided a solid foundation for addressing some of the technical challenges in this domain. Despite these advancements, the reviewed literature predominantly emphasizes static strategies for resource allocation and task scheduling, with limited attention to the dynamic interplay between cloud and edge resources [9–11]. Moreover, there is a notable absence of comprehensive frameworks that consider real-time resource utilization and workload dynamics in orchestration decisions, as well as a lack of studies addressing the dynamic nature of edge environments and their implications for resource allocation and task distribution [12, 13]. These limitations underscore the necessity for research that not only bridges these gaps but also introduces adaptive orchestration strategies that can accommodate changing network conditions and workload patterns in cloud-edge environments. The key contributions of this study are as follows.

- Utilize the PRISMA model to methodically scrutinize and reduce 10,389 records to 89 high-quality studies.
- Identify challenges in resource management and task allocation, highlighting opportunities for enhanced service delivery.
- Evaluate the performance of cloud-edge orchestration using the MAPE-K framework.
- Discuss task distribution strategies to balance over- and under-provisioning in cloud and edge

environments.
- Categorize performance metrics to optimize resource collaboration and system efficiency in cloud-edge orchestration.
- Offer insights for improving cloud-edge integration to fulfil organizational operational needs.

The ensuing sections of this study are organized in the following sequence: Section II culminates in a comprehensive assessment of the significant outcomes of related studies. Section III meticulously outlines the review procedure employed in this systematic literature review (SLR), providing in-depth details on the study retrieval and selection process. Section IV presents the results of carefully selected studies. In Section V, the studies included in the review offer valuable insights and perspectives into the selected RQs, as displayed in Table I. Finally, the conclusion and future work of this research are summarized in Section VI.

II. Related Study

Cloud and edge orchestration have been extensively studied, with different approaches addressing various challenges in resource allocation and performance optimization. Castellano *et al.* [14] proposed a service-defined orchestration framework that allows applications to specify their own strategies, offering flexibility in cloud-edge resource management. However, their study assumed a homogeneous infrastructure, overlooking the complexities of heterogeneous environments and potential security risks. In contrast, Mittal *et al.* [15] introduced a vehicular-based task orchestration framework that leverages vehicle-to-vehicle communication to create dynamic vehicular edges. Although innovative, their approach suffers from high latency and connection instability owing to vehicular mobility and network fluctuations, making it less reliable for real-time applications. Similarly, Caballer *et al.* [16] explored the coordination of multiple Infrastructure-as-a-Service (IaaS) resources using open-source tools such as OpenStack and TOSCA. Despite demonstrating the feasibility of multi-cloud orchestration, their study highlighted significant interoperability challenges and lacked an in-depth evaluation of performance bottlenecks when integrating different cloud providers.

Wu [17] examined AI-driven cloud-edge orchestration for IoT applications, enabling adaptive data processing and collaboration between cloud and edge environments. However, their approach is constrained by the limited computational and storage capacity of IoT devices, which can hinder the effectiveness of AI-driven orchestration and fail to address energy efficiency concerns, which is an essential factor in IoT deployments. Although each study presents valuable insights into orchestration strategies, limitations such as security vulnerabilities, scalability concerns, network instability, and resource constraints remain key obstacles, suggesting the need for more robust and adaptive orchestration mechanisms that can effectively balance flexibility, performance, and reliability in heterogeneous cloud-edge environments. The studies presented in the three subheadings encompass diverse aspects of cloud-edge orchestration techniques, with particular emphasis on workload balancing, virtualization techniques, and edge resource management. A comparative analysis of these studies, including their respective limitations, reveals the following.

A. *Workload Balancing*

Kim [18] and Ding *et al.* [19] employed cooperative game theory for workload distribution and resource optimization. Although these approaches demonstrate the potential for efficient resource allocation and workload balancing across cloud and edge environments, they may face scalability challenges in large-scale dynamic environments with numerous devices and rapidly fluctuating workloads. Akhlaqi and Hanapi [20] introduced a predictive workload balancing model that utilized machine learning techniques to analyze historical workload patterns and optimize resource distribution. Although this method improves load forecasting, it suffers from a high computational overhead, making it less practical for real-time applications in cloud-edge environments. To address this issue, Bao *et al.* [21] integrated deep learning with workload balancing frameworks to enable the real-time prediction of workload spikes and proactive resource allocation. However, deep learning models introduce computational complexity, making them less suitable for resource-constrained edge devices. Cerroni *et al.* [22] conducted a comparative analysis of cloud-based service performance against an analytical model, highlighting response time limitations for commercial routers. Although this study provides valuable insights into performance constraints, its focus on specific device types may limit its applicability to the diverse range of edge devices in real-world applications.

B. *Virtualization Techniques*

Valsamas *et al.* [23] and Wang *et al.* [24] explored Virtualization Technology Blending (VTB) to enhance resource utilization and orchestration efficiency. Although this approach shows promise for improving user capacity and resource utilization, the studies may be limited in their consideration of the overhead introduced by blending multiple virtualization technologies and the potential impact on system complexity. Ren *et al.* [25] concentrated on distributed edge computing and compared new computational frameworks. This study highlights the necessity of further research to investigate other computing paradigms beyond the present monopoly of cloud computing. Zhang *et al.* [26] proposed a chunk reuse mechanism (CRM) to optimize container updates. Although this approach can reduce data transmission and improve efficiency, it may face challenges in scenarios with highly diverse or rapidly changing application requirements, potentially limiting its effectiveness in specific edge computing environments.

C. *Edge Resource Management*

Chiang *et al.* [27] provided a comprehensive analysis of service orchestration and resource management, evaluating multiple performance metrics. Although this study offers valuable insights, the rapidly evolving nature of edge computing technologies may necessitate frequent updates to maintain their relevance. Gharbaoui *et al.* [28] evaluated SDN-based orchestration for improving VM allocation and resource utilization. This approach demonstrates potential for enhancing efficiency but may encounter challenges in environments with limited SDN support or in scenarios where the overhead of SDN implementation outweighs its benefits. Li *et al.* [29] proposed an energy-efficient solution that optimizes containerized workflow scheduling to enhance deployment efficiency while minimizing energy consumption. Despite its advantages, this approach lacks predictive workload balancing capabilities, leading to occasional resource bottlenecks. Soumplis *et al.* [30] presented methods using mixed-integer linear programming, a greedy algorithm, and a multi-agent rollout mechanism. Although these approaches offer diverse solutions, they may be constrained by computational complexity in large-scale deployments or scenarios with strict real-time requirements.

Workload balancing studies focus on optimizing task distribution and resource allocation, whereas virtualization techniques aim to enhance flexibility and efficiency in resource usage. Edge resource management studies address the challenges of limited resources at the edge while maintaining the quality of service. Cooperative game theory approaches to workload balancing (Kim [18], Ding *et al.* [19]) offer sophisticated optimization techniques but may face scalability challenges. In contrast, virtualization techniques such as VTB (Valsamas *et al.* [23], Wang *et al.* [24]), and CRM (Zhang *et al.* [26]) provide more direct improvements in resource utilization but may introduce additional complexity. Edge resource management studies (Chiang *et al.* [27] and Gharbaoui *et al.* [28], Soumplis *et al.* [30]) offer a broader perspective on orchestration challenges, considering multiple performance metrics and exploring diverse algorithmic approaches. However, these studies may face limitations in real-world applications owing to the dynamic nature of edge computing environments. The main contributions of the different studies are presented in Table II.

TABLE II: A COMPILATION OF RESEARCH SUMMARIES ON THE ORCHESTRATION TECHNIQUES

Aspects	Study	Orchestration technique	Performance indicators	Limitations
Workload-balancing	Kim [18]	Cooperative game theory for workload distribution	Resource allocation efficiency, Workload balance	Scalability challenges in large-scale, dynamic environments
	Cerroni *et al.* [22]	Comparative analysis of cloud-based service performance	Response time	Limited applicability to diverse edge devices
	Ding *et al.* [19]	Cooperative game theory for resource optimization	Resource allocation efficiency, Workload balance	Scalability challenges in large-scale, dynamic environments
Virtualization Techniques	Zhang *et al.* [26]	Chunk reuse mechanism (CRM) for container updates	Data transmission reduction, Update efficiency	Limited effectiveness in scenarios with diverse or rapidly changing application requirements
	Valsamas *et al.* [23]	Virtualization Technology Blending (VTB)	User capacity, Resource utilization	Potential overhead and increased system complexity
	Wang *et al.* [24]	Virtualization Technology Blending (VTB)	Resource utilization, Orchestration efficiency	Potential overhead and increased system complexity
Edge Resource Management	Chiang *et al.* [27]	Comprehensive analysis of service orchestration and resource management	Latency, Throughput	May require frequent updates to maintain relevance
	Soumplis *et al.* [30]	Mixed-integer linear programming, Greedy algorithm, multi-agent rollout mechanism	Resource allocation efficiency	Computational complexity in large-scale deployments or strict real-time scenarios
	Gharbaoui *et al.* [28]	SDN-based orchestration	VM allocation efficiency, Resource utilization	Challenges in environments with limited SDN support

III. REVIEW PROCEDURE

The procedures required for conducting an SLR are outlined in [31], which we have used as an outline for this study. To systematically organize and document the selection process, we employed the preferred reporting items for systematic reviews and meta-analyses (PRISMA) model, which ensures a transparent and replicable methodology for conducting systematic reviews. This SLR was divided into three phases, as shown in Fig. 1.

1) Planning phase

In this initial stage, the RQs were clearly articulated and a comprehensive review strategy was formulated. This phase established the foundation for a systematic and unbiased selection process.

2) Conducting phase

This phase involved the selection of primary studies for review, ensuring a rigorous and unbiased approach through the following steps.

- *Search Strategy:* A well-defined search strategy was designed to identify relevant primary research using appropriate search queries and reliable data sources.
- *Selection Standards:* Studies were selected based on predefined inclusion and exclusion criteria, ensuring systematic differentiation between the included and excluded studies.
- *Quality Indicators*: The quality of the selected studies was assessed using established benchmarks to ensure their reliability and relevance.
- *PRISMA:* The PRISMA framework was applied to document the study selection process, including identification, screening, eligibility assessment, and final inclusion in this review.

3) Reporting phase

In this final phase, the research questions were systematically addressed based on the findings of the selected studies, ensuring a transparent and comprehensive synthesis of the results.

The PRISMA model helps maintain rigor and clarity in systematic reviews, thereby enhancing reproducibility and credibility.

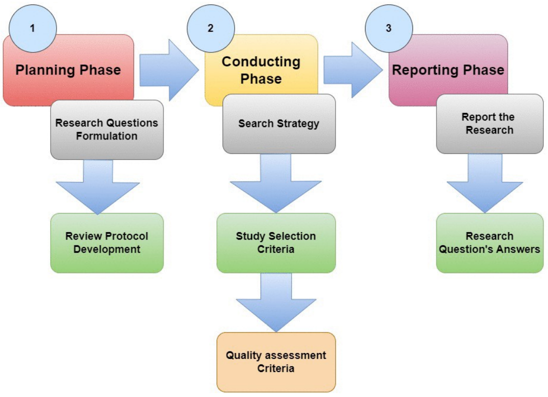
1
Planning Phase
2
Conducting Phase
3
Reporting Phase
Research Questions Formulation
Search Strategy
Report the Research
Review Protocol Development
Study Selection Criteria
Research Question's Answers
Quality assessment Criteria

Fig. 1. An outline of the systematic literature review procedure.

A. Search Strategy

To ensure a comprehensive and systematic literature review, we formulated targeted search phrases to identify relevant studies on edge-cloud orchestration within the MAPE-K framework. These search phrases were strategically constructed using Boolean operators (“AND” and “OR”) to enhance the retrieval of relevant publications. The primary search terms were categorized as follows: (“cloud” OR “edge” OR “collaboration”), (“monitoring” OR “analysis” OR “planning” OR “execution”), AND (“resource provisioning” OR “task allocation”). Once the search phrases were verified, we selected six prestigious electronic databases for literature retrieval, ensuring coverage across high-impact academic sources. The chosen databases included the following:

- Science Direct
- Springer Link
- Wiley Online Library
- IEEE Xplore Digital Library
- ACM Digital Library
- Taylor & Francis

Using these databases, we conducted a structured search of research published within the past decade (2015–2024).

B. Selection Standards

To narrow down the number of publications and examine the most pertinent ones, this systematic review employed the inclusion and exclusion standards, outlined in Table III. First, search queries containing titles, abstracts, and keywords were included to select the related studies. Second, only studies conducted after 2015 were included. Third, the most cited publications were considered, and fourth this work includes papers pertinent to the topic. As a result, the list of included publications has to be derived from a systematic search and contain novel or intriguing concepts related to the review topic. Conversely, a set of exclusion standards was developed to eliminate studies that were incompatible with the purpose of this SLR. Research not published in English was excluded. Less comprehensive research related to duplicate publications was disregarded. Articles published in conferences, symposiums, and workshops were excluded. Furthermore, publications that were unambiguously unrelated to the primary subject matter were excluded. After applying all exclusion standards (ES1–ES4), 89 publications were included in this evaluation.

TABLE III: SELECTION CRITERIA

Inclusion Standards (IS)		Exclusion Standards (ES)	
IS1	Identification by the search queries	ES1	Publication not in English
IS2	Publications between 2015 to 2024	ES2	Duplicate and irrelevant scope publications
IS3	Most cited papers	ES3	Publications in conferences, symposiums, and workshops
IS4	Studies relevant to the topic	ES4	Publisher not aligned with the study

C. Quality Indicators (QIs)

To ensure the rigor and validity of the selected studies, we conducted a comprehensive quality assessment based on a set of well-defined quality indicators (QIs), as outlined in Table IV. Each study was meticulously evaluated against these indicators to determine its relevance, methodological robustness, and potential impact on our research. The PRISMA flow diagram was used to systematically filter and assess studies, ensuring that only those meeting the predefined quality criteria were included in the final selection. Five quality indicators (QI1–QI5) were employed to assess the suitability of each study.

TABLE IV: QUALITY ASSESSMENT RUBRIC

ID	QIs	Response
QI1	Adherence to academic integrity standards through citations	Influential contributions to the literature
QI2	Applicability of research findings	The relevance of the study to the present endeavor
QI3	A precise description of the objective	An explicit goal for the study
QI4	Explicit methodology	An in-depth delineation of the applicable model, assignment of task criteria during execution, performance metrics, and monitoring, analysis, planning, and execution techniques.
QI5	Contrastive analysis	Incorporation of a comparative assessment of relevant literature

- *Academic Integrity and Citation Influence (QI1):* The extent to which the study adheres to academic integrity standards through proper citations and its contribution to the scholarly discourse.
- *Applicability of Research Findings (QI2):* The relevance of the study’s findings in addressing the research objectives of the present investigation.
- *Clarity of Research Objectives (QI3):* The presence of a well-defined research aim ensures a precise focus on the study's intended contribution.
- *Explicit Methodological Framework (QI4):* The comprehensiveness of the study’s methodology, including the delineation of task distribution mechanisms, task allocation strategies, performance metrics, and the MAPE-K framework.
- *Comparative Analysis (QI5):* The inclusion of a

contrastive assessment that contextualizes findings within existing literature, demonstrating thorough engagement with previous studies.

D. PRISMA-Based Literature Selection Process

To systematically organize and document the selection process, we employed the PRISMA model, which enhances methodological rigor, transparency, and reproducibility in systematic reviews. The PRISMA framework structures the literature selection into four key phases: identification, screening, eligibility, and inclusion.

- *Identification Phase:* A total of 10,389 records were retrieved from the selected databases. At this stage, 1,039 duplicate records (ES2) were automatically removed, and an additional 519 records were excluded based on the predefined ineligibility criteria. This filtering resulted in 8,831 records proceeding to the screening phase.
- *Screening Phase:* The remaining 8,831 records were manually screened. A total of 4,416 records were excluded as conference or workshop publications (ES3). Consequently, 4,415 reports were retrieved. However, due to access restrictions and irrelevance, 662 reports could not be retrieved, leaving 3,753 reports for full-text assessment.
- *Eligibility Phase:* A total of 3,753 full-text articles were evaluated for relevance based on predefined inclusion criteria. During this phase, 2,627 records were excluded because of non-English language publications (ES1) and publisher misalignment (ES4), reducing the selection pool to 1,126 full-text articles. A further 1,014 articles were excluded after a detailed assessment, ensuring that only high-relevance studies progressed to the final stage.
- *Inclusion Phase:* Ultimately, 99 studies were selected for qualitative synthesis, comprising 89 journal studies and 10 SLRs.

Following the PRISMA framework, this study ensures a systematic, unbiased, and reproducible review process, enabling a comprehensive evaluation of cloud-edge orchestration methodologies in cloud-edge computing environments. Fig. 2 presents a visual representation of the systematic review process, illustrating the structured approach employed in this study using the PRISMA flowchart.

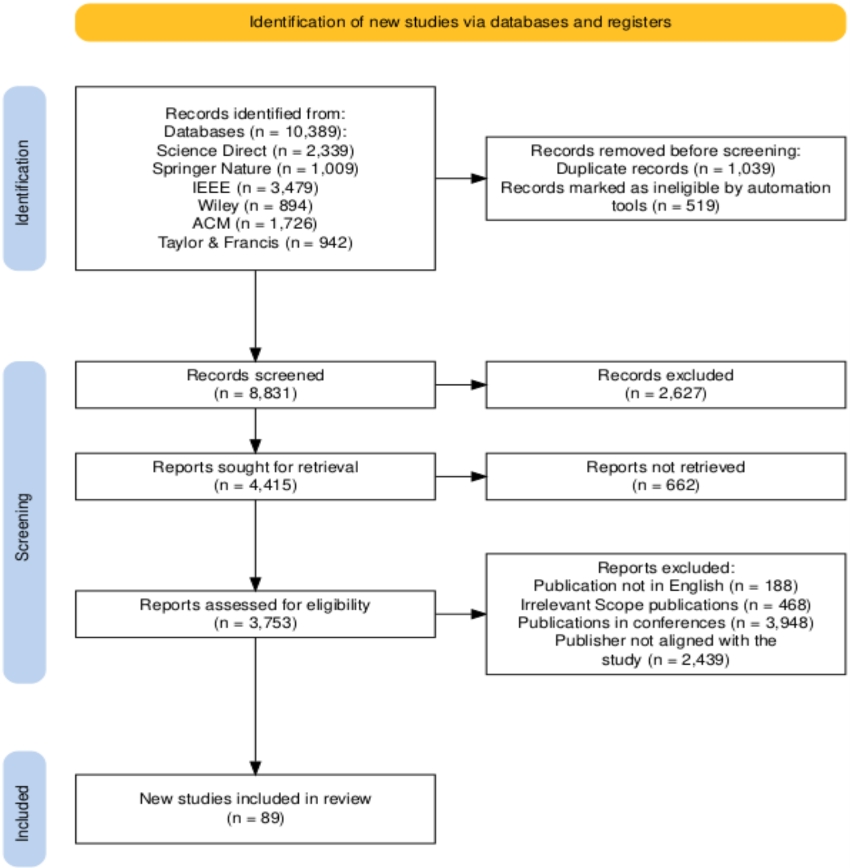
Identification of new studies via databases and registers
Identification
Records identified from:
Databases (n = 10,389):
Science Direct (n = 2,339)
Springer Nature (n = 1,009)
IEEE (n = 3,479)
Wiley (n = 894)
ACM (n = 1,726)
Taylor & Francis (n = 942)
Records removed before screening:
Duplicate records (n = 1,039)
Records marked as ineligible by automation tools (n = 519)
Screening
Records screened (n = 8,831)
Records excluded (n = 2,627)
Reports sought for retrieval (n = 4,415)
Reports not retrieved (n = 662)
Reports assessed for eligibility (n = 3,753)
Reports excluded:
Publication not in English (n = 188)
Irrelevant Scope publications (n = 468)
Publications in conferences (n = 3,948)
Publisher not aligned with the study (n = 2,439)
Included
New studies included in review (n = 89)

Fig. 2. Systematic review roadmap via PRISMA.

IV. Results

This comprehensive SLR explores the evolving domain of edge-cloud orchestration, focusing on key aspects from the MAPE-K perspective, including resource provisioning and allocation. A rigorous search strategy was employed to identify relevant studies, ensuring alignment with the study objectives through carefully formulated search keywords. The literature search was conducted across six prominent digital repositories—Science Direct, IEEE Xplore, SpringerLink, Wiley Online Library, ACM Digital Library, and Taylor

& Francis—selected for their strong academic credibility and extensive research coverage in the field.

The initial search retrieved a total of 10,389 research articles, distributed as follows: IEEE Xplore (3,479), Science Direct (2,339), ACM Digital Library (1,726), SpringerLink (1,009), Wiley Online Library (894), and Taylor & Francis (942). To ensure the inclusion of only the most relevant studies, strict selection criteria were applied, leading to the final inclusion of 89 studies: 21 from ScienceDirect, 15 from IEEE Xplore, 19 from SpringerLink, 13 from Wiley Online Library, 13 from ACM Digital Library, and 8 from Taylor & Francis. Furthermore, using quality indicator metrics, we identified 10 SLRs and 89 journal publications for an in-depth analysis of the literature.

Additional analyses were conducted to provide a comprehensive overview of publication trends. The publication timeline graph illustrates the progression of research output over the past decade, while the source distribution graph offers insights into the dissemination of knowledge across various academic communities and underscores the increasing scholarly attention toward cloud-edge orchestration.

A. Publication Time

The chosen studies were across-verified against the predetermined criteria to guarantee the validity of the results. These studies were conducted in the past decade. Notably, the highest number of publications were recorded in 2022 and 2023. There has been a noticeable increase in research in 2023 compared with previous years. Fig. 3 shows that the number of studies from 2015 to 2024 were 2, 2, 3, 4, 12, 10, 8, 18, 22, and 8, respectively. The observed trend in the number of publications indicates that this research area has garnered significant attention in recent years and exhibited a notable growth trajectory. This suggests a promising future scope for further exploration in this field.

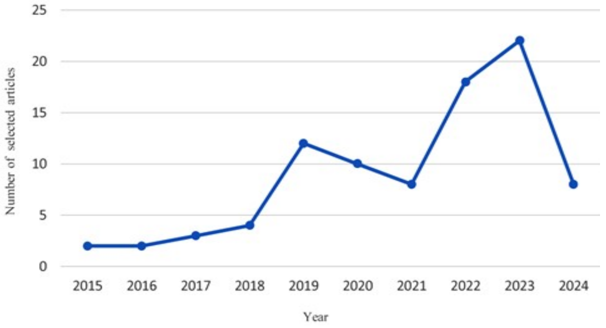

Fig. 3. Studies retrieved from each year.

B. Publication Sources

Fig. 4 presents the distribution of the selected articles across the six major electronic databases for this review. Among these sources, ScienceDirect contributed the highest number of selected papers (21), followed by SpringerLink (19), IEEE Xplore (15), ACM Digital Library (13), Wiley Online Library (13), and Taylor and Francis (8). Notably, SpringerLink and ScienceDirect emerged as the most significant contributors to this review, highlighting their extensive coverage of cloud-edge orchestration research.

Additionally, a key observation from IEEE Xplore and ACM Digital Library is the substantial presence of IEEE/ACM Transaction papers, which play a crucial role in advancing research in this field. Specifically, we selected 15 papers from IEEE Xplore and 13 from the ACM Digital Library, of which 16 are IEEE/ACM Transactions. These transaction papers are highly regarded for their rigorous peer-review processes and significant contributions to theoretical advancements, system architectures, and performance evaluations. Their inclusion enhances the credibility and depth of this SLR by providing well-validated insights into cloud-edge orchestration and ensuring a comprehensive understanding of recent developments.

This distribution highlights the varying degrees of relevance and coverage of the selected research topics across different digital repositories. The presence of high-quality contributions across multiple sources reinforces the importance of diverse literature in capturing the full spectrum of research on edge-cloud orchestration and resource management.

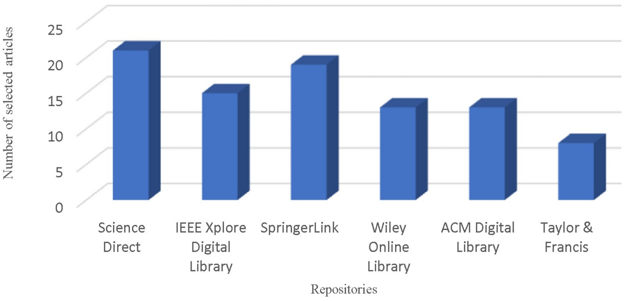

Fig. 4. Contributions from diverse sources.

V. Discussion

This section presents a comprehensive evaluation of the studies included in this review, meticulously categorizing their outcomes in alignment with the RQs in Table II. Section A delves into the MAPE-K approach, a tool employed to assess the performance of cloud-edge orchestration techniques. Subsequently, Section B outlines the pertinent performance metrics for orchestration within collaborative cloud-edge environments. Section C describes the task distribution between cloud and edge computing as determined during execution. Section D discusses outstanding issues and difficulties encountered. The prospects and emerging trends in cloud-edge computing are finally revealed in Section E.

A. To What Extent Does The MAPE-K Framework Effectively Assess the Performance of Collaborative Cloud-Edge Techniques? (RQ1)

The MAPE-K approach coordinates the monitoring, analysis, planning, and execution of the task distribution processes. The knowledge base of the framework stores information about the edge and cloud resources, task requirements, and the performance metrics. It is used in cloud and edge computing to enable distributed decision-making. This loop comprises four primary components: monitoring, analysis, planning, and execution. The monitoring component extracts data from the edge nodes of the network. It collects data on the resource usage of edge and cloud assets, such as "CPU utilization," "memory usage," and "network bandwidth" usage. It analyzes resource usage data to identify potential over- or under-provisioning of resources. The data were then analyzed to determine the state of the environment at the time and to identify possible areas for improvement or optimization. It evaluates the effectiveness of the current task allocation techniques using the performance metrics retained in the knowledge base. It determines tasks that are not being processed efficiently (such as a CPU-intensive activity being executed on a resource with limited CPU capacity) or executed on the incorrect resource (such as latency-sensitive work being executed on the cloud). Subsequently, a new task distribution plan was developed using a knowledge database that balanced resource provisioning. Finally, actions are taken based on these plans to validate the plan's efficacy. The execution component considers the task requirements, resource utilization, and performance measures when executing task between the cloud and edge. It enables dynamic decision-making interactions among network nodes, as depicted in Fig. 5. As all nodes can interact, making decisions collaboratively without depending on the controller is easy to achieve.

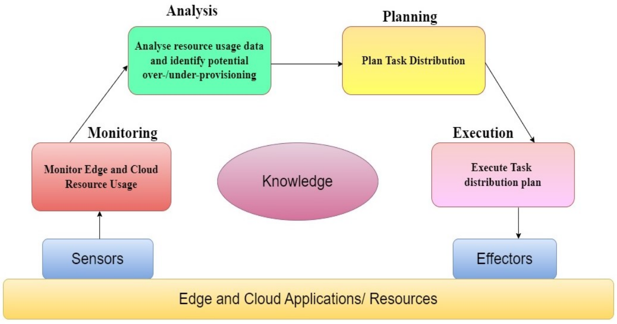
Analysis
Analyse resource usage data and identify potential over-/under-provisioning
Planning
Plan Task Distribution
Monitoring
Monitor Edge and Cloud Resource Usage
Knowledge
Execution
Execute Task distribution plan
Sensors
Effectors
Edge and Cloud Applications/ Resources

Fig. 5. Dynamic decision-making loop.

1) *Monitoring*

Monitoring systematically reviews, perceives, and controls the operational workflows in cloud-based system infrastructures. It emphasizes different aspects and techniques in a cloud-edge context. It gathers data from the surrounding environment and compares them with expectations. This allows system performance to be evaluated, allowing for the identification of changes and corrections to maintain the system performance. CloudWatch, a service offered by AWS, is a well-known example [32]. Manual or automated management strategies ensure the reliability and efficiency of Internet platforms, servers, software applications, and other cloud platforms. Researchers have proposed monitoring frameworks and architectures primarily for cloud-edge orchestration to monitor and manage data flows, resource usage, performance measures, and security [33]. These studies considered using monitoring tools and techniques to monitor and analyze data in real-time, recognize performance bottlenecks, determine security threats, and improve resource allocation [34–36]. Table V summarizes the work done in the field of monitoring.

TABLE V: RELATED STUDIES ON MONITORING

Study	Key Findings	Focus	Limitations
[36]	Monitor the performance of cloud-edge applications with high accuracy	Performance monitoring	Limited accuracy
[37]	Provide comprehensive security monitoring with broader coverage	Security monitoring	Limited coverage
[38]	Reduce data collection latency and improve scalability	Data collection	Limited scalability
[39]	Reduce data storage overhead and improve data access speed	Data storage	High overhead
[40]	Collect data while preserving privacy with acceptable latency overhead	Privacy-preserving data collection	High latency
[41]	Detect anomalies in cloud-edge collaboration with a low false positive rate.	Anomaly detection	High false positive rate
[42]	Enable self-monitoring in collaborative cloud and edge computing with reduced complexity.	Self-monitoring	High complexity
[43]	Enable federated learning with improved communication efficiency	Federated learning	Limited communication efficiency
[44]	Allocate resources efficiently with reduced computational overhead	Resource allocation	High computational overhead
[45]	Schedule tasks adaptably to improve resource utilization	Task scheduling	Limited adaptability

2) *Analysis*

Analysis is the process of understanding the performance of a system after implementation. It is performed on collected information to check whether it fulfills the system's goals [46]. It examines anomalies and discrepancies that occur during the execution of the system and then performs the corresponding corrective measures to fix them. In a hybrid cloud environment, data are processed in the cloud and at the edge, close to the data source [47]. This enables real-time processing and analysis to take place. Various technologies have been explored for cloud-edge orchestration [48, 49]. Fantacci and Picano [50] proposed a federated learning framework for Mobile Edge Computing (MEC) networks that enables multitasking in a distributed environment. Tefera *et al.* [51] have explored multi-access edge computing networks and also worked on congestion-aware adaptive decentralized computational offloading to control high traffic issues in edge computing networks. Recent studies [52–54] described privacy preservation using machine learning and data analytic methodologies, respectively, in cloud-edge contexts, whereas Tang *et al.* [55] examined anomaly detection using a distributed knowledge distillation framework in a cloud-edge scenario. The developments that have taken place in the analysis field are listed in Table VI.

TABLE VI: RELATED STUDIES ON THE ANALYSIS

Study	Key Findings	Focus	Limitations
[51]	Develop a federated learning framework for federated multitask learning in cloud-edge	Federated learning, multitask learning, cloud-edge computing	Increased complexity in coordinating and managing

	contexts.		multiple tasks in a distributed setting
[52]	Design a real-time stream processing framework for low-latency data analysis in collaborative cloud and edge.	Real-time stream processing, Cloud-Edge computing	Challenges in handling high-volume and high-velocity data streams
[53]	Propose a federated learning approach for privacy-preserving using machine learning in cloud-edge environments.	Federated learning, Privacy-preservation, and machine learning	Increased communication overhead compared to traditional centralized machine learning
[54]	Design a privacy-preserving edge analytic framework for secure data processing in IoT networks.	Privacy-preserving data analysis, Edge computing, and IoT networks	Increased computational overhead due to privacy-enhancing mechanisms
[55]	Develop a distributed machine learning framework for real-time anomaly detection in IoT sensor data.	Edge computing, and Distributed machine learning	Limited scalability for large-scale IoT deployments

3) *Planning*

Effective planning is crucial for the design and analysis of a system. Proper planning helps resolve issues that have been identified after the analysis of data or systems [56]. It optimally utilizes the resources shared between the cloud and edge nodes and delivers data services to consumers in a timely manner. A comprehensive body of literature has explored various aspects of hybrid cloud systems [57–59]. For instance, Mach and Beevar [60] examined the effect of planning on the performance of mobile devices using application offloading in MEC. The authors proposed a power control system for real-time applications that considers the performance and energy consumption of edge devices. In contrast, Wang *et al.* [61] devised a solution for the scalability of dynamic workloads in multi-tenant edge environments. Furthermore, for the optimal distribution of tasks, Durga *et al.* [62] proposed a data-flow-driven mechanism for global heterogeneous systems, whereas Vinothkumar *et al.* [63] designed an energy-efficient and reliable data collection mechanism using IoT and smart grids. Moreover, recent studies have explored different aspects of security and scaling techniques in cloud-edge computing contexts. Lampropoulos and Siakas [64] reviewed security and privacy, while Dogani *et al.* [65] surveyed auto-scaling techniques in container-based cloud and edge computing. A representation of the most recent research in the planning field is depicted in Table VII.

TABLE VII: RELATED STUDIES ON PLANNING

Study	Key Findings	Focus	Limitations
[61]	Investigating the impact of planning on edge device performance and battery life	Edge device profiling, performance optimization, and energy consumption analysis	Challenges in accurately modelling device performance and battery life
[62]	Optimizing data placement and movement for cost-efficiency and latency minimization	Cloud-edge resource allocation, data partitioning, and data replication	Uncertainty in workloads and network conditions
[63]	Designing energy-efficient data planning algorithms for edge devices	Energy consumption minimization, task scheduling, and resource utilization	Heterogeneity of edge devices and data formats
[64]	Improving data privacy and security in cloud-edge environments	Secure data storage. Access control, data encryption	Balancing security with performance and cost-efficiency
[65]	Examining auto-scaling approaches for anticipating future resource demands and workload patterns	Scaling techniques, resource forecasting, and adaptive data management	Complexity of predictive models and uncertainty in future directions
[66]	Developing scalable planning solutions for dynamic cloud edge computing environments	Scalability, workload adaptation, and resource optimization	Real-time requirements and uncertainty in resource availability

4) *Execution*

After collecting information from IoT devices, it is analyzed, planned, and executed according to consumers' requirements. Several investigations have been conducted in the direction of execution in an integrated cloud-edge context [67–69]. To optimally utilize resources and maintain the real-time performance of cloud-edge computing, Maenhaut *et al.* [70] reviewed resource management in a containerized cloud system, whereas Du *et al.* [71] explored data placement strategies to maximize resource utilization and minimize latency in heterogeneous edge-cloud computing systems. Rawashdeh *et al.* [72] proposed a security framework for the quantum-as-a-service (QaaS) model. This model was designed to process massive amounts of data in an intelligent transport system that provides improved decision-making and predictions. Gajmal and Udayakumar [73] explored privacy-and utility-assisted data protection strategies for secure data sharing and retrieval in cloud-edge systems. Additionally, Zhou *et al.* [74] reviewed the existing literature on production and operation administration for intelligent manufacturing and outlined crucial issues and research directions. Bao and Guo [75] proposed a federated learning framework to maintain privacy in collaborative cloud-edge systems. Moreover, Chen *et al.* [76] exploited a deep reinforcement approach for optimal task scheduling in a collaborative cloud-edge environment. The most recent investigation on execution in a cloud-edge scenario is presented in Table VIII.

TABLE VIII: RELATED STUDIES ON EXECUTION

Study	Key Findings	Focus	Limitations
[70]	Explore resource management techniques for cloud-edge environments that optimize resource utilization and ensure real-time performance.	Resource management in cloud-edge environments	Lack of real-world deployments and evaluation
[71]	Propose a dynamic data placement strategy for hybrid cloud environments to minimize latency and maximize resource	Data partitioning and placement	Heterogeneity of hardware and software platforms

	utilization.		
[72]	Design a secure data consistency mechanism for cloud-edge contexts that guarantees data integrity while minimizing communication overhead.	Security and fault tolerance	Dynamic and unpredictable workloads
[73]	Investigate security and privacy issues in cloud-edge collaboration and propose a secure data execution framework.	Security and privacy	The complexity of managing security and privacy across multiple domains
[74]	Survey the existing literature on production and operation management in collaborative cloud edge and identify key challenges and research directions.	Overview of data execution in collaborative cloud-edge environments	Lack of comprehensive solutions for all challenges
[75]	Propose a federated learning framework for cloud-edge environments to enable collaborative machine learning without compromising privacy.	Federated learning in cloud-edge environments	The limited scalability of federated learning algorithms
[76]	Develop a task scheduling algorithm for cloud-edge contexts, considering latency and resource constraints.	Task scheduling and resource allocation	Unreliable and bandwidth-constrained network connectivity

B. How Can We Measure the Performance of Cloud-Edge Orchestration? (RQ2)

The following section first articulates the categorization of quantifiable performance measures according to the MAPE-K loop, and then their mapping corresponds to the existing literature on cloud-edge computing.

1) Nomenclature of performance metrics

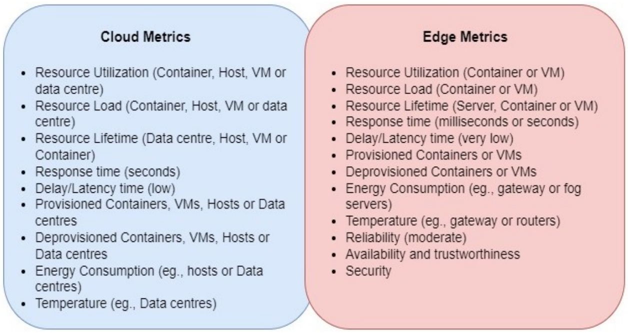
Cloud Metrics
Resource Utilization (Container, Host, VM or data centre)
Resource Load (Container, Host, VM or data centre)
Resource Lifetime (Data centre, Host, VM or Container)
Response time (seconds)
Delay/Latency time (low)
Provisioned Containers, VMs, Hosts or Data centres
Deprovisioned Containers, VMs, Hosts or Data centres
Energy Consumption (eg., hosts or Data centres)
Temperature (eg., Data centres)
Edge Metrics
Resource Utilization (Container or VM)
Resource Load (Container or VM)
Resource Lifetime (Server, Container or VM)
Response time (milliseconds or seconds)
Delay/Latency time (very low)
Provisioned Containers or VMs
Deprovisioned Containers or VMs
Energy Consumption (eg., gateway or fog servers)
Temperature (eg., gateway or routers)
Reliability (moderate)
Availability and trustworthiness
Security

Fig. 6. Categorization of performance metrics for evaluating cloud-edge orchestration.

The performance metrics of an IoT application depend on the computing model in which it is organized. For example, in edge computing, the resiliency of IoT appliances in executing offloaded tasks is an essential hurdle because of their limited computation capacity [32]. In contrast, resource availability is guaranteed in cloud computing. However, each performance statistic is considered to have a different extent. Edge computing approaches adopt some traits from centralized computing to resemble the potential performance measures. Specific metrics are standard on the edge and cloud, whereas others are classified separately according to particular requirements. For instance, resource usage can be assessed in terms of containers and VMs, depending on the relevant cloud. This inspired the creation of Fig. 6, which presents a taxonomy of the identified performance measures for the collaborative cloud.

- *Monitoring metrics*

Various monitoring metrics, such as resource utilization, response time, QoS, and service level agreement (SLA), energy consumption, latency, and throughput, play vital roles in cloud-edge orchestration. Resource utilization metrics measure the deployment of physical and virtual resources [77]. The response time is the total time required to complete a task. This helps to recognize and uncover particular inefficiencies to improve processing time. Metrics related to QoS and SLA measure the quality of service and service level agreement a company provides to the customer according to their requirements. Energy consumption is the amount of energy utilized by a system over time. Moreover, the latency metric measures the time required for data packet transmission between the sender and receiver, whereas throughput is used to calculate the data transfer rate over time.

- *Analysis metrics*

Analyzing metrics is used to predict the monitoring parameters. Most of these metrics are prediction- and time-series-based and can be observed using various machine-learning methods. These metrics are based on statistical analysis measurements, which include several techniques for collecting and analyzing data to identify the trends, patterns, and relationships. The statistical analysis included descriptive methods such as mean, median, and variance, and inferential analysis methods such as Analysis Of Variance (ANOVA) and t-tests. Moreover, it includes prescriptive analysis methods such as simulation, graph analysis, and algorithms; predictive analysis techniques such as data mining, modelling, and artificial intelligence; causal analysis techniques such as software quality assurance; exploratory analysis methods such as hypothesis testing, finding errors; and mechanistic analysis.

- *Planning metrics*

Planning is fundamentally based on optimization decisions, such as VM placement and migration. This was performed after monitoring and analysis. It involves several metrics, such as orchestration decisions, contradictory decisions, scalability, and competition ratio, which play significant roles in the planning domain. Orchestration decision metrics make scaling, computations, and application decisions. It decides which application should run at the edge and which should be sent back to the cloud. These decisions rely on the prevailing network infrastructure and resources in collaborative cloud-edge computing environments. Contradictory decisions refer to overturning already made decisions owing to errors in those decisions. After the decisions are applied, the scalability metrics measure the time taken to attain a balanced and productive state. These fall under the general category of adaptation time, which displays the potential capacity of the orchestration to be compressed and respond to the workload. This phenomenon can be observed when the quantity of resources in a cloud environment or edge infrastructure increases. Moreover, another metric, that is, the competition ratio, describes it as the proportion of resources competing to fulfil a request or vice versa.

- *Execution metrics*

Execution occurs after monitoring, analysis, and planning. This is the ultimate stage, during which orchestration operations are performed. Several metrics are related to execution, such as orchestration actions, provisioned decisions, deprovisioning decisions, cost, profit, security, and privacy. The first metric is the orchestration of actions relevant to the automation and administration of the cloud infrastructure. This metric involves setting, installing, and administrating resources such as VMs, database servers, and storage platforms. The second metric is the provisioned decision metric, also known as the deployment metric. These are resources allocated to specific tasks or applications throughout the runtime. Deprovisioning metrics are defined as the number of resources released when they are not required for a long period. The cost metric is defined as the monetary or complexity cost that the user or service provider experiences during the runtime of an orchestration technique. The next metric is profit, which is defined as the revenue retained by the effective utilization of the orchestration approach. Finally, the security and privacy metric can be described as the orchestration's ability to preserve information and authorized amenities from unauthorized access.

2) *Mapping of performance metrics*

The measures outlined in the literature for optimizing collaborative cloud edge computing environments are listed in Table IX. The spatial distribution of metrics in the table indicates which measures are most prominent. The most commonly evaluated metrics are the efficient use of resources, response time, energy expenditure, and cost, which are all measurable via the monitor in the MAPE-K loop. The diversity of analyzer-related measures, which are essentially statistical metrics, is more constrained. Planner-related metrics are essential because contradictory decisions can significantly affect the orchestration performance, even though the executor phase frequently overrides such choices. Cost is a well-studied metric in the cloud computing literature. Although conducting cost assessments in such a dynamic and federated context is challenging, their evaluation in cloud-edge orchestration could be more comprehensive in the future. The monitor and executor are the primary evaluation points for most metrics because of their connections with the outside world.

TABLE IX: MAPPING OF PRELIMINARY TAXONOMY WITH THE CONTEMPORARY LITERATURE

Performance Evaluation Metrics	Study																
	[78]	[79]	[80]	[81]	[82]	[83]	[84]	[85]	[86]	[87]	[7]	[88]	[89]	[90]	[91]	[92]	[93]
Resource Utilization	√	√			√				√	√	√		√	√	√	√	
Task response time	√	√	√	√		√		√	√					√	√		
QoS/ SLA			√	√								√			√	√	
Energy Consumption				√		√		√	√			√	√			√	
Latency			√					√			√		√		√	√	√
Throughput													√			√	
Statistical Analysis Measurements																√	
Orchestration Decision					√											√	
Contradictory Decision														√			
Scalability					√								√		√		√
Competition Ratio																	√
Orchestration Actions					√										√		
Provisioned Resources					√										√		
Deprovisioned Resources					√										√		
Cost	√	√		√	√	√				√			√		√	√	
Profit											√		√		√	√	
Security & Privacy							√					√				√	√

C. *What Mechanisms Determine the Optimal Distribution of Tasks Between Cloud and Edge Nodes? (RQ 3)*

Edge computing is a propitious approach that exploits the potential of resource-intensive applications on IoT devices by moving computing power to cloud infrastructure. However, restricted computing resources are a significant liability. It is difficult to determine the ideal computing power for edge computing and the optimal number of servers to be installed at the edge. Finding a balance between over- and under-provisioning is vital for addressing cloud adoption issues. To access optimal task distribution mechanisms, the following section discusses resource balancing between cloud and edge nodes and the task distribution strategies. The allocation of more resources than required results in over-provisioning, whereas the allocation of fewer resources results in under-provisioning. It is not easy to maintain QoS while balancing over- and under-provisioning. The possible solutions to balance between these two are as follows: (i) Deploy abundant fixed resources by increasing the number of servers at the edge node. (ii) Collaborating resources between different clouds.

1) *Deployment of Abundant Resources*

The main benefit of deploying abundant fixed resources is that it will ease the handling of overwhelming workloads. However, estimating the abundance of these resources is challenging. In addition, deploying abundant resources wastes resource costs and resource utilization. Furthermore, it may contend with the issue of both over-and under-provisioning. Therefore, instead of deploying a fixed amount of hardware resources, which involves a high probability of resource wastage, the alternate option is to deploy a few hardware resources on the edge server and tenant the rest from the cloud server [94]. The benefits of cloud-edge orchestration are twofold: (i) the probability of loss of profit decreases, and (ii) the probability of resource wastage also decreases (because the cloud resources are available on a demand basis as well as on a pay-per-use basis). Fig. 7. represents the architecture of a cloud-edge collaboration.

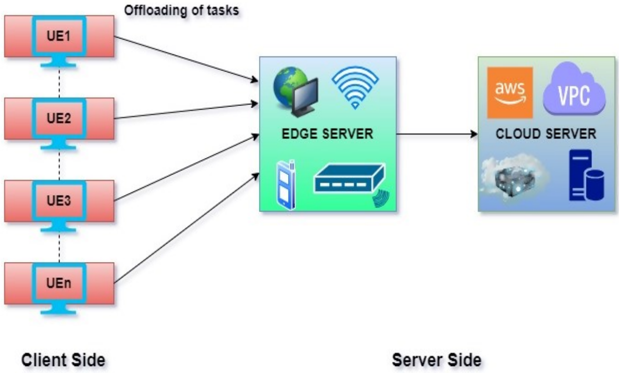

Fig. 7. Cloud-edge collaboration architecture.

2) *Collaboration between cloud and edge using VMware strategies*

One of the most prevalent cloud-based services is VM provisioning, especially because cloud computing stipulates a broad spectrum of attributes, allowing Cloud Service Providers (CSPs) and Cloud Service Users (CSUs) to adapt their computational capacity to meet operational requirements [95]. CSUs can procure VM instances through spot bidding, on-demand, or reservations. Contractual agreements (often 1 to 3 years) are required to purchase reserved VM instances, which have a high maintenance cost, although pervasive consumption of reserved VMs is also accessible via huge discounts [96]. Reserved VMs with extensive consumption can reduce the total expenditure by as much as 38% annually. Similarly, saving 60% of the total expense is possible by reserving VMs for three years. However, the varying workload and unpredictability of the shortened request wait times sometimes render it undesirable to acquire all VMs on a reservation basis.

In contrast, on-demand instances of VMs can be purchased whenever required. Compared to reserved VM instances, the cost of per-time quantum tenancy for on-demand VMs is more significant than usual. If a VM instance is only used for a few hours per day or a couple of days per month, on-demand instances are the ideal substitute [97]. Online auctioning is an alternative for procuring spot instances; however, it does not ensure availability. Therefore, this study did not examine such instances. When assessing the effects of leasing plans, we contemplated the possibility of a Software-as-a-Service (SaaS) provider because a CSU may seldom use its reserved VMs. This provides an alternative to traditional leasing, wherein customers can use the service on demand. In this way, the CSU can purchase a certain amount of computing power and pay only for the amount it uses. SaaS providers lease VMs from infrastructure providers, such as Amazon EC2, to host IoT device services [98]. In load fluctuation (varying workload demand by users), cloud service providers use a short-term rental model for temporary needs based on the on-demand VM's pay-per-hour usage.

D. *What Are the Open Issues and Preliminary Challenges of Cloud-Edge Orchestration? (RQ4)*

Cloud computing is an astonishing technology that has the potential to provide a range of assets to its tenants; however, it can be limited by the computing resources in edge computing. To overcome this limitation, tenants can lease resources from cloud data centers [95]. However, this arrangement poses challenges. These challenges include the costs associated with leasing resources from remote locations and ensuring compatibility between resources from different providers. It is crucial to consider these challenges when deciding on the necessary resources for an application to ensure optimal performance. This section highlights the associated challenges that must be addressed before adopting edge-cloud collaboration for mainstream use.

1) *Distribution of tasks between edge and cloud*

Contemporary researchers advocate different task distribution techniques for overwhelming workloads. Some advocate that delay-sensitive tasks should be offloaded to edge computing, assuming that it has sufficient computing resources [99]. However, edge computing resources can also suffer under-provisioning during the overwhelming workload of delay-sensitive tasks. Some studies have suggested complex optimization models for the distribution of tasks between the edge and cloud, using offline scheduling algorithms to achieve different objectives. For instance, overall waiting time reduction tasks, overall makespan reduction, latency minimization, and efficiency maximization. However, the critical issue with such systems is deciding when to execute such optimization models, that is, every minute or every hour. In addition, it would unnecessarily increase the waiting time for certain tasks. Moreover, it would require prioritization of tasks, which would induce indiscrimination. Similarly, some approaches suggest multilevel feedback queues for reducing the starvation of delay-tolerant tasks [100]. However, a critical issue with such systems is when to upgrade the priority status of delay-tolerant tasks.

2) *Measuring the impact of offloading on system performance*

Edge computing aims to optimize the perks of offloading by serving as an alternative to conventional centralized computing. However, precisely assessing the benefits of offloading is difficult because of multiple factors. These include differences in the transmission rate of the network, variations in the processing capacity of remote VMs, and fluctuations in the assortment of instructions in the task. Furthermore, the edge may offer distinct VM types and cloud-based service providers, and the performance variations of these machines may not be equally pronounced.

In a virtual box, we performed basic experiments using two different types of VMs to illustrate the impact of different VM sizes [101]. The first VM had a CPU speed of 3.2 GHz, whereas the second one has a speed of 2.5 GHz. We examined an example application, bubble sort, with a worst-case time complexity of O(n^2). Fig. 8 shows the execution times for both VMs using four sets of random numbers. According to the trial results, there was no significant difference in the performance of the two VMs. This indicates that there is no 28% increase in the execution time of the 3.2 GHz CPU virtual machine over the 2.5 GHz core CPU machine. However, deep learning and other applications with exponential execution times exacerbate this issue. Hence, rigorous tools should be designed to estimate the performance of tasks on VMs so that users can be assured of the offloading benefits.

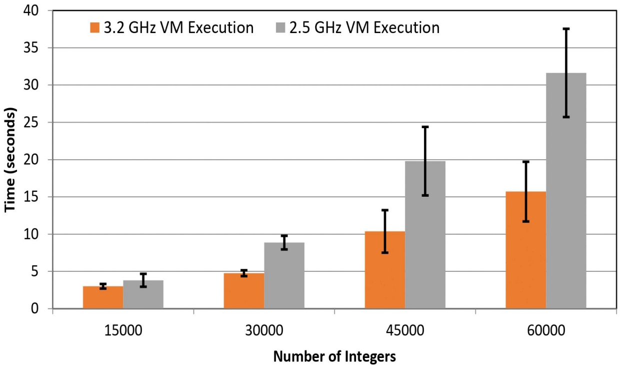

Fig. 8. Execution time of bubble sort on different size virtual machines.

3) Pricing model

The tenanting costs of resources can vary among different cloud infrastructure service providers. For instance, Amazon provides its virtual machine instances in reserved, on-demand, and spot bidding forms [98]. In addition, these instances are available at per-ten-minute or per-hour lease costs. On-demand instances can be acquired without upfront costs or long-term commitments. In contrast, reserved instances require high upfront costs and long-term obligations (e.g., three years). Consequently, such instances are available at discounted rates compared to the on-demand instances. However, the under-utilization of reserved instances can cost more than on-demand instances because of the fixed monthly charges. Therefore, deciding on an optimal leasing plan is challenging.

4) Privacy

The collaborative dissemination of dynamic data between edge and cloud data centers increases the probability of cloud vulnerabilities, as cloud data centers may be located in different countries with different security and privacy policies.

5) Mobility management

Data processing is hindered when a user moves from the proximity of one data center to another. In this case, tracking the processing status of the user's task and resuming it at another data center is not easy. The European Telecommunications Standards Institute (ETSI) standard advocates three task relocation schemes [85]. First, task state relocation suggests stopping and resuming the task at the target location. The second approach relocates every possible volunteer task to the intended server within the MEC. Meanwhile, the task remains active on the original server until it is transferred. Third, the mechanism is optional and feasible only with cloud support. Thus, rigorous efforts must be made to provide seamless services during the move.

6) Resource requirement estimation

The edge service providers can lower leasing costs by utilizing various auto-scaling mechanisms, such as response time-based auto-scaling systems, threshold-rule-based auto-scaling methods, or trade-offs between services and waiting costs-based approaches [102]. However, such systems are either application-specific or service-level agreement-specific. Moreover, unpredictable workload arrivals can degrade the performance of auto-scaling systems. In addition, the significant setup time of VMs would further aggravate the miseries of the edge service providers because the service provider has to deal with the existing machines during VMs. Hence, it is challenging to deal with unpredictable workloads when the VM startup time is significant for maintaining the QoS for end users.

E. What Are the Opportunities and Future Trends in Cloud-Edge Orchestration? (RQ5)

Despite these challenges, there remains a small window of opportunity. Some of them are covered in this subsection.

1) Expansion of effective performance measures

A significant hurdle in the orchestration between edge and cloud environments is the creation of effective measures to evaluate the effectiveness of optimization techniques. This is crucial because it allows practitioners and researchers to evaluate different optimization strategies and adjudicate the most effective strategy for a particular application. Numerous aspects must be considered when implementing performance metrics for edge and cloud orchestration. Although researchers have developed several performance measures in this direction, no commonly accepted measure has been established. Therefore, further research is required to develop and standardize edge and cloud orchestration performance metrics.

2) Assessing novel approaches to resource allocation and scheduling

The integration of cloud and edge computing faces several challenges, such as latency issues in cloud computing, the distributed nature and limited computational capacity of edge computing, and the dynamic nature of IoT workloads. To resolve these issues, experts are exploring new techniques such as deep reinforcement learning, game theory algorithms, and federated learning algorithms. These unique algorithms improve the effectiveness and performance of collaborative cloud-edge systems.

3) Employment of secure and private data distribution policies

Collaborative clouds depend on data security and privacy. It is essential to protect personal information collected and processed by edge devices from unauthorized access. There is a higher chance of eavesdropping when data are shared between cloud and edge devices. Researchers are developing various strategies to transfer sensitive data between the edge and the cloud. More efforts are required to evolve and optimize secure and private data transfer mechanisms to facilitate cloud-edge orchestration. Moreover, researchers must establish standards and guidelines for the secure and private use of cloud-edge orchestration.

4) Deployment and management tools

Various frameworks and tools, such as edge orchestration platforms, cloud-edge data management tools, and edge-cloud security solutions, are required to implement and maintain cloud-edge orchestration systems. They enable the automation of many complex tasks in the establishment and management of orchestration of edge-cloud systems. They also support a broader range of edge devices and applications. Furthermore, new frameworks and technologies have been specifically designed to address several edge-cloud orchestration applications. By developing new and improved frameworks and tools for edge-cloud cooperation, we can make it easier for users to deploy, manage, and benefit from this exciting new paradigm.

VI. Conclusion and Future Work

Cloud-edge orchestration involves the management and collaboration of resources and tasks across the cloud and edge computing systems. It optimizes the distribution and management of data and processes across edge and cloud environments, resulting in enhanced performance and reliability. It is becoming a more widespread technology in this rapidly changing digital world because it integrates the advantages of edge computing, such as low latency and real-time performance, with cloud computing, which offers enormous processing capabilities. The primary objective of this SLR is to provide insights into the adoption of cloud-edge orchestration techniques in both industry and academia. We meticulously explored the systematic literature by utilizing the search terms listed in Fig. 2 and the PRISMA model to conduct an extensive review, resulting in 10,389 records from the past ten years. These were then narrowed down to 89 studies and 10 SLRs. We provide an in-depth discussion of the research questions listed in Table I. The selected studies extensively examined several aspects of cloud-edge orchestration, such as resource provisioning, monitoring, analysis, planning, execution, performance assessment, and scheduling.

This SLR provides valuable insight and diverse perspectives on the benefits and drawbacks of cloud and edge computing systems. We systematically assessed the study through an extensive evaluation using a structured review protocol. We briefly discuss a compilation of research summaries on collaborative edge-cloud computing environments. Additionally, we utilized the MAPE-K approach to evaluate the efficacy of cloud-edge orchestration methodologies and examined advancements in monitoring, analysis, planning, and execution across various fields. Furthermore, we categorized and mapped performance metrics to provide an extensive review of contemporary studies. To address the issue of over-and under-provisioning, we discussed the criteria for task scheduling between edge and cloud. Finally, we identify and discuss the associated challenges and propose future directions.

As technology advances, cloud-edge orchestration is becoming increasingly popular in the technical industry and holds significant potential for further advancements in computing. Based on the outcomes of the research questions, this SLR concludes that collaboration between cloud and edge through orchestration techniques faces persistent challenges. To effectively alleviate these challenges, comprehensive and accurate performance metrics must be examined. Additionally, further research is needed on resource allocation, task distribution, and security considerations. In future work, to enhance the performance and efficiency in cloud-edge orchestration, we will concentrate on execution domain exploration and investigate the optimal way to distribute tasks across cloud and edge nodes.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Nisha Saini conducted the research, wrote the paper and analyzed the data. Jitender Kumar analyzed the results and reviewed the work. All authors approved the final version.

References

[1] K. Walia, M. Kumar, and S. S. Gill, "AI-empowered fog/edge resource management for IoT applications: A comprehensive review, research challenges and future perspectives," *IEEE Communications Surveys & Tutorials*, 2023. doi: 10.1109/COMST.2023.3338015

[2] P. Varshney and Y. Simmhan, "Characterizing application scheduling on edge, fog, and cloud computing resources," *Softw. Pract. Exp.*, vol. 50, no. 5, pp. 558–595, May 2020.

[3] E. Huedo, R. S. Montero, R. Moreno-Vozmediano, C. Vázquez, V. Holer, and I. M. Llorente, "Opportunistic deployment of distributed edge clouds for latency-critical applications," *Journal of Grid Computing*, vol. 19, pp. 1–16, 2021. doi: 10.1007/s10723-021-09545-3

[4] G. Rong, Y. Xu, X. Tong, and H. Fan, "An edge-cloud collaborative computing platform for building AIoT applications efficiently," *J. Cloud Comp.*, vol. 10, no. 1, Dec. 2021. doi: 10.1186/s13677-021-00250-w

[5] S. Shahzadi, M. Iqbal, T. Dagiuklas, and Z. U. Qayyum, "Multi-access edge computing: open issues, challenges and future perspectives," *J. Cloud Comp.*, vol. 6, no. 1, Dec. 2017. doi: 10.1186/s13677-017-0097-9

[6] J. Liu, E. Ahmed, M. Shiraz, A. Gani, R. Buyya, and A. Qureshi, "Application partitioning algorithms in mobile cloud computing: Taxonomy, review and future directions," *Journal of Network and Computer Applications*, vol. 48, pp. 99–117, 2015. doi: 10.1016/j.jnca.2014.09.009

[7] W. Z. Khan, E. Ahmed, S. Hakak, I. Yaqoob, and A. Ahmed, "Edge computing: A survey," *Future Generation Computer Systems*, vol. 97, pp. 219–235, 2019. doi: 10.1016/j.future.2019.02.050

[8] Y. Miao, G. Wu, M. Li, A. Ghoneim, M. Al-Rakhami, and M. S. Hossain, "Intelligent task prediction and computation offloading based on mobile-edge cloud computing," *Future Generation*

Computer Systems, vol. 102, pp. 925–931, 2020. doi: 10.1016/j.future.2019.09.035

[9] Y. Mansouri and M. A. Babar, "A review of edge computing: Features and resource virtualization," *Journal of Parallel and Distributed Computing*, vol. 150, pp. 155–183, 2021. doi: 10.1016/j.jpdc.2020.12.015

[10] A. Shakarami, H. Shakarami, M. Ghobaei-Arani, E. Nikougoftar, and M. Faraji-Mehmandar, "Resource provisioning in edge/fog computing: A comprehensive and systematic review," *Journal of Systems Architecture*, vol. 122, 102362, 2022. doi: 10.1016/j.sysarc.2021.102362

[11] A. Mampage, S. Karunasekera, and R. Buyya, "A holistic view on resource management in serverless computing environments: taxonomy and future directions," *ACM Comput. Surv.*, vol. 54, no. 11s, pp. 1–36, Jan. 2022.

[12] Y. Laili, F. Guo, L. Ren, X. Li, Y. Li, and L. Zhang, "Parallel scheduling of large-scale tasks for industrial cloud–edge collaboration," *IEEE Internet of Things Journal*, vol. 10, no. 4, pp. 3231–3242, 2021.

[13] M. Raeisi-Varzaneh, O. Dakkak, A. Habbal, and B.-S. Kim, "Resource scheduling in edge computing: Architecture, taxonomy, open issues and future research directions," *IEEE Access*, vol. 11, pp. 25329–25350, 2023. doi: 10.1109/ACCESS.2023.3256522

[14] G. Castellano, F. Esposito, and F. Risso, "A service-defined approach for orchestration of heterogeneous applications in cloud/edge platforms," *IEEE Trans. on Network and Service Management*, vol. 16, no. 4, pp. 1404–1418, 2019.

[15] S. Mittal, R. K. Dudeja, R. S. Bali, and G. S. Aujla, "A distributed task orchestration scheme in collaborative vehicular cloud edge networks," *Computing*, vol. 106, no. 4, pp. 1151–1175, 2024.

[16] M. Caballer, S. Zala, Á. L. García, G. Moltó, P. O. Fernández, and M. Velten, "Orchestrating complex application architectures in heterogeneous clouds," *Journal of Grid Computing*, vol. 16, pp. 3–18, 2018. doi: 10.1007/s10723-017-9418-y

[17] Y. Wu, "Cloud-edge orchestration for the Internet of Things: Architecture and AI-powered data processing," *IEEE Internet of Things Journal*, vol. 8, no. 16, pp. 12792–12805, 2020.

[18] S. Kim, "Collaborative resource sharing game based cloud-edge offload computing orchestration scheme," *IEEE Access*, vol. 10, pp. 74523–74532, 2022. doi: 10.1109/ACCESS.2022.3190857

[19] Y. Ding, K. Li, C. Liu, and K. Li, "A potential game theoretic approach to computation offloading strategy optimization in end-edge-cloud computing," *IEEE Trans. on Parallel and Distributed Systems*, vol. 33, no. 6, pp. 1503–1519, 2021.

[20] M. Y. Akhlaqi and Z. B. M. Hanapi, "Task offloading paradigm in mobile edge computing-current issues, adopted approaches, and future directions," *Journal of Network and Computer Applications*, vol. 212, 103568, 2023. doi: 10.1016/j.jnca.2022.103568

[21] Y. Bao, Y. Peng, and C. Wu, "Deep learning-based job placement in distributed machine learning clusters with heterogeneous workloads," *IEEE/ACM Trans. on Networking*, vol. 31, no. 2, pp. 634–647, 2022.

[22] W. Cerroni, M. Gharbaoui, B. Martini, A. Campi, P. Castoldi, and F. Callegati, "Cross-layer resource orchestration for cloud service delivery: A seamless SDN approach," *Computer Networks*, vol. 87, pp. 16–32, 2015. doi: 10.1016/j.comnet.2015.05.008

[23] P. Valsamas, S. Skaperas, L. Mamatas, and L. M. Contreras, "Virtualization technology blending for resource-efficient edge clouds," *Computer Networks*, vol. 225, 109646, 2023. doi: 10.1016/j.comnet.2023.109646

[24] Q. Wang, D. Gao, C. H. Foh, H. Zhang, and V. C. Leung, "Protocols design and area division for privacy-preserving delay-aware authentication in vehicular networks," *IEEE Trans. on Vehicular Technology*, vol. 70, no. 11, pp. 11129–11144, 2021.

[25] J. Ren, D. Zhang, S. He, Y. Zhang, and T. Li, "A survey on end-edge-cloud orchestrated network computing paradigms: Transparent computing, mobile edge computing, fog computing, and cloudlet," *ACM Computing Surveys (CSUR)*, vol. 52, no. 6, pp. 1–36, 2019.

[26] H. Zhang, W. Lin, R. Xie, S. Li, Z. Dai, and J. Z. Wang, "An optimal container update method for edge-cloud collaboration," *Softw Pract Exp*, vol. 54, no. 4, pp. 617–634, Apr. 2024.

[27] Y. Chiang, Y. Zhang, H. Luo *et al.*, "Management and orchestration of edge computing for IoT: A comprehensive survey," *IEEE Internet of Things Journal*, vol. 10, no. 16, pp. 14307–14331, 2023.

[28] M. Gharbaoui, B. Martini, D. Adami, S. Giordano, and P. Castoldi, "Cloud and network orchestration in SDN data centers: Design principles and performance evaluation," *Computer Networks*, vol. 108, pp. 279–295, 2016. doi: 10.1016/j.comnet.2016.08.029

[29] F. Li, W. J. Tan, and W. Cai, "A wholistic optimization of containerized workflow scheduling and deployment in the cloud–edge environment," *Simulation Modelling Practice and Theory*, vol. 118, 102521, Jul. 2022. doi: 10.1016/j.simpat.2022.102521

[30] P. Soumplis, G. Kontosv, P. Kokkinos, A. Kretsis, S. Barrachina-Muñoz, R. Nikbakht, J. Baranda, M. Payaró, J. Mangues-Bafalluy, and E. Varvarigos, "Performance optimization across the edge-cloud continuum: A multi-agent rollout approach for cloud-native application workload placement," *SN Comput. Sci.*, vol. 5, no. 3, p. 318, Mar. 2024. doi: 10.1007/s42979-024-02630-w

[31] D. Budgen and P. Brereton, "Evolution of secondary studies in software engineering," *Information and Software Technology*, vol. 145, 106840, 2022. doi: 10.1016/j.infsof.2022.106840

[32] M. S. Aslanpour, S. S. Gill, and A. N. Toosi, "Performance evaluation metrics for cloud, fog and edge computing: A review, taxonomy, benchmarks and standards for future research," *Internet of Things*, vol. 12, 2020. doi: 10.1016/j.iot.2020.100273

[33] H. Yang, S. K. Ong, A. Y. C. Nee, G. Jiang, and X. Mei, "Microservices-based cloud-edge collaborative condition monitoring platform for smart manufacturing systems," *International Journal of Production Research*, vol. 60, no. 24, pp. 7492–7501, Dec. 2022.

[34] C. Liu, P. Zheng, and X. Xu, "Digitalisation and servitisation of machine tools in the era of Industry 4.0: A review," *International Journal of Production Research*, vol. 61, no. 12, pp. 4069–4101, Jun. 2023.

[35] Z. Amiri, A. Heidari, N. J. Navimipour, and M. Unal, "Resilient and dependability management in distributed environments: A systematic and comprehensive literature review," *Cluster Computing*, vol. 26, no. 2, pp. 1565–1600, 2023.

[36] J. Soldani and A. Brogi, "Anomaly detection and failure root cause analysis in (Micro) service-based cloud applications: A survey," *ACM Comput. Surv.*, vol. 55, no. 3, pp. 1–39, Mar. 2023.

[37] S. Taherizadeh, A. C. Jones, I. Taylor, Z. Zhao, and V. Stankovski, "Monitoring self-adaptive applications within edge computing frameworks: A state-of-the-art review," *Journal of Systems and Software*, vol. 136, 2018. doi: 10.1016/j.jss.2017.10.033

[38] G. Shin, M. H. Jarrahi, Y. Fei, A. Karami, N. Gafinowitz, A. Byun, and X. Lu, "Wearable activity trackers, accuracy, adoption, acceptance and health impact: A systematic literature review," *Journal of biomedical informatics*, vol. 93, 103153, 2019. doi: 10.1016/j.jbi.2019.103153

[39] A. S. Veith, M. D. Assuncao, and L. Lefevre, "Latency-aware strategies for deploying data stream processing applications on large cloud-edge infrastructure," *IEEE Trans. on Cloud Computing*, 2021. doi: 10.1109/TCC.2021.3097879

[40] Q. Liang, W. A. Hanafy, A. Ali-Eldin, and P. Shenoy, "Model-driven cluster resource management for AI workloads in edge clouds," *ACM Trans. Auton. Adapt. Syst.*, vol. 18, no. 1, pp. 1–26, Mar. 2023.

[41] C. Jian, J. Ping, and M. Zhang, "A cloud edge-based two-level hybrid scheduling learning model in cloud manufacturing," *International Journal of Production Research*, vol. 59, no. 16, pp. 4836–4850, Aug. 2021.

[42] J. Zhang, Z. Qu, C. Chen, H. Wang, Y. Zhan, B. Ye, and S. Guo, "Edge Learning: The Enabling Technology for Distributed Big Data Analytics in the Edge," *ACM Comput. Surv.*, vol. 54, no. 7, pp. 1–36, Sep. 2022. doi: 10.1145/3464419

[43] X. Xu, W. Liu, Y. Zhang, X. Zhang, W. Dou, L. Qi, and M.Z.A. Bhuiyan, "PSDF: Privacy-aware IoV service deployment with federated learning in cloud-edge computing," *ACM Trans. Intell. Syst. Technol.*, vol. 13, no. 5, pp. 1–22, Oct. 2022.

[44] S. Mihai, M. Yaqoob, D. V. Hung *et al.*, "Digital twins: A survey on enabling technologies, challenges, trends and future prospects," *IEEE Communications Surveys & Tutorials*, vol. 24, no. 4, pp. 2255–2291, 2022.

[45] Y. Li, B. Liu, E. Wu, J. Li, Z. Zhou, and W. Zhang, "DRA-MQoS: An MQoS scheduling algorithm based on resource feature matching in federated edge cloud," *Concurrency and Computation*, vol. 35, no. 2, p. e7478, Jan. 2023. doi: 10.1002/cpe.7478

[46] A. Yousafzai, A. Gani, R. M. Noor *et al.*, "Cloud resource allocation schemes: review, taxonomy, and opportunities,"

Knowledge and Information Systems, vol. 50, pp. 347–381, 2017. doi: 10.1007/s10115-016-0951-y

[47] D. Rosendo, A. Costan, P. Valduriez, and G. Antoniu, "Distributed intelligence on the Edge-to-Cloud Continuum: A systematic literature review," *Journal of Parallel and Distributed Computing*, vol. 166, 2022. doi: 10.1016/j.jpdc.2022.04.004

[48] R. Ghosh and Y. Simmhan, "Distributed scheduling of event analytics across edge and cloud," *ACM Trans. Cyber-Phys. Syst.*, vol. 2, no. 4, pp. 1–28, Oct. 2018.

[49] T. L. Duc, R. G. Leiva, P. Casari, and P.-O. Östberg, "Machine learning methods for reliable resource provisioning in edge-cloud computing: A survey," *ACM Comput. Surv.*, vol. 52, no. 5, pp. 1–39, Sep. 2020.

[50] R. Fantacci and B. Picano, "Federated learning framework for mobile edge computing networks," *CAAI Trans. Intell. Technol.*, vol. 5, no. 1, pp. 15–21, Mar. 2020.

[51] G. Tefera, K. She, M. Chen, and A. Ahmed, "Congestion-aware adaptive decentralised computation offloading and caching for multi-access edge computing networks," *IET Communications*, vol. 14, no. 19, pp. 3410–3419, Dec. 2020.

[52] Y. Zhao, Z. Yang, X. He, X. Cai, X. Miao, and Q. Ma, "Trine: Cloud-edge-device cooperated real-time video analysis for household applications," *IEEE Trans. on Mobile Computing*, vol. 22, no. 8, pp. 4973–4985, 2022.

[53] M. Hosseinzadeh, A. Hemmati, and A. M. Rahmani, "Federated learning-based IoT: A systematic literature review," *Int. J. Communication*, vol. 35, no. 11, 2022. doi: 10.1002/dac.5185

[54] A. K. Nair, J. Sahoo, and E. D. Raj, "Privacy preserving Federated Learning framework for IoMT based big data analysis using edge computing," *Computer Standards & Interfaces*, vol. 86, 103720, 2023. doi: 10.1016/j.csi.2023.103720

[55] L. Tang, C. Xue, Y. Zhao, and Q. Chen, "Anomaly detection of service function chain based on distributed knowledge distillation framework in cloud-edge industrial internet of things scenarios," *IEEE Internet of Things Journal*, 2023. doi: 10.1109/JIOT.2023.3327795

[56] S. M. N. A. Sunny, X. "Frank" Liu, and M. R. Shahriar, "Development of machine tool communication method and its edge middleware for cyber-physical manufacturing systems," *International Journal of Computer Integrated Manufacturing*, vol. 36, no. 7, pp. 1009–1030, Jul. 2023.

[57] C. Li, H. Sun, H. Tang, and Y. Luo, "Adaptive resource allocation based on the billing granularity in edge-cloud architecture," *Computer Communications*, vol. 145, pp. 29–42, 2019. doi: 10.1016/j.comcom.2019.05.014

[58] C.-H. Hong and B. Varghese, "Resource management in fog/edge computing: A survey on architectures, infrastructure, and algorithms," *ACM Comput. Surv.*, vol. 52, no. 5, pp. 1–37, Sep. 2020.

[59] B. B. Gupta, A. Gaurav, P. K. Panigrahi, and V. Arya, "Analysis of cutting-edge technologies for enterprise information system and management," *Enterprise Information Systems*, vol. 17, no. 11, p. 2197406, Nov. 2023. doi: 10.1080/17517575.2023.2197406

[60] P. Mach and Z. Becvar, "Cloud-aware power control for real-time application offloading in mobile edge computing," *Trans. Emerging Tel. Tech.*, vol. 27, no. 5, pp. 648–661, May 2016.

[61] N. Wang, M. Matthaiou, D. S. Nikolopoulos, and B. Varghese, "DYVERSE: dynamic vertical scaling in multi-tenant edge environments," *Future Generation Computer Systems*, vol. 108, pp. 598–612, 2020. doi: 10.1016/j.future.2020.02.043

[62] S. Durga, E. Daniel, J. A. Onesimu, and Y. Sei, "Resource provisioning techniques in multi-access edge computing environments: outlook, expression, and beyond," *Mobile Information Systems*, pp. 1–24, Dec. 2022.

[63] T. Vinothkumar, S. S. Sivaraju, A. Thangavelu, and S. Srithar, "An energy-efficient and reliable data gathering infrastructure using the internet of things and smart grids," *Automatika: časopis za automatiku, mjerenje, elektroniku, računarstvo i komunikacije*, vol. 64, no. 4, pp. 720–732, 2023.

[64] G. Lampropoulos and K. Siakas, "Enhancing and securing cyber-physical systems and Industry 4.0 through digital twins: A critical review," *J. Software Evolu. Process*, vol. 35, no. 7, p. e2494, Jul. 2023. doi: 10.1002/smr.2494

[65] J. Dogani, R. Namvar, and F. Khunjush, "Auto-scaling techniques in container-based cloud and edge/fog computing: Taxonomy and survey," *Computer Communications*, vol. 209, pp. 120–150, 2023. doi: 10.1016/j.comcom.2023.06.010

[66] T. H. Son, Z. Weedon, T. Yigitcanlar, T. Sanchez, J. M. Corchado, and R. Mehmood, "Algorithmic urban planning for smart and sustainable development: Systematic review of the literature," *Sustainable Cities and Society*, vol. 94, 104562, 2023. doi: 10.1016/j.scs.2023.104562

[67] K. Peng, H. Huang, S. Wan, and V. C. Leung, "End-edge-cloud collaborative computation offloading for multiple mobile users in heterogeneous edge-server environment," *Wireless Networks*, pp. 1–12, 2020. doi: 10.1007/s11276-020-02385-1

[68] R. Jeyaraj, A. Balasubramaniam, A. K. M.A., N. Guizani, and A. Paul, "Resource management in cloud and cloud-influenced technologies for internet of things applications," *ACM Comput. Surv.*, vol. 55, no. 12, pp. 1–37, Dec. 2023.

[69] H. Hou, S. N. A. Jawaddi, and A. Ismail, "Energy efficient task scheduling based on deep reinforcement learning in cloud environment: A specialized review," *Future Generation Computer Systems*, vol. 151, pp. 214–231, 2024. doi: 10.1016/j.future.2023.10.002

[70] P.-J. Maenhaut, B. Volckaert, V. Ongenae, and F. De Turck, "Resource management in a containerized cloud: Status and challenges," *Journal of Network and Systems Management*, vol. 28, pp. 197–246, 2020. doi: 10.1007/s10922-019-09504-0

[71] X. Du, S. Tang, Z. Lu, K. Gai, J. Wu, and P. C. K. Hung, "Scientific workflows in IoT environments: A data placement strategy based on heterogeneous edge-cloud computing," *ACM Trans. Manage. Inf. Syst.*, vol. 13, no. 4, pp. 1–26, Dec. 2022.

[72] M. Rawashdeh, Y. Alshboul, M. G. A. Zamil, S. Samarah, A. Alnusair, and M. S. Hossain, "A security framework for QaaS model in intelligent transportation systems," *Microprocessors and Microsystems*, vol. 90, 104500, 2022. doi: 10.1016/j.micpro.2022.104500

[73] Y. M. Gajmal and R. Udayakumar, "Privacy and utility-assisted data protection strategy for secure data sharing and retrieval in cloud system," *Information Security Journal: A Global Perspective*, vol. 31, no. 4, pp. 451–465, Jul. 2022.

[74] L. Zhou, Z. Jiang, N. Geng, Y. Niu, F. Cui, K. Liu, and N. Qi, "Production and operations management for intelligent manufacturing: a systematic literature review," *International Journal of Production Research*, vol. 60, no. 2, pp. 808–846, 2022.

[75] G. Bao and P. Guo, "Federated learning in cloud-edge collaborative architecture: key technologies, applications and challenges," *J. Cloud Comp.*, vol. 11, no. 1, p. 94, Dec. 2022.

[76] Z. Chen, L. Zhang, X. Wang, and K. Wang, "Cloud–edge collaboration task scheduling in cloud manufacturing: An attention-based deep reinforcement learning approach," *Computers & Industrial Engineering*, vol. 177, 109053, 2023. doi: 10.1016/j.cie.2023.109053

[77] P. Neelakantan and N. S. Yadav, "An optimized load balancing strategy for an enhancement of cloud computing environment," *Wireless. Pers. Commun.*, vol. 131, no. 3, pp. 1745–1765, 2023.

[78] A. M. Senthil Kumar and M. Venkatesan, "Multi-objective task scheduling using hybrid genetic-ant colony optimization algorithm in cloud environment," *Wireless Pers Commun*, vol. 107, no. 4, pp. 1835–1848, Aug. 2019.

[79] Y. Xing, "Work scheduling in cloud network based on deep Q-LSTM models for efficient resource utilization," *J. Grid Computing*, vol. 22, no. 1, p. 36, Mar. 2024.

[80] A. Aral, I. Brandic, R. B. Uriarte, R. De Nicola, and V. Scoca, "Addressing application latency requirements through edge scheduling," *J. Grid Computing*, vol. 17, no. 4, pp. 677–698, 2019.

[81] A. Ahmed, S. Azizi, and S. R. M. Zeebaree, "ECQ: An energy-efficient, cost-effective and qos-aware method for dynamic service migration in mobile edge computing systems," *Wireless Pers. Commun.*, vol. 133, no. 4, pp. 2467–2501, Dec. 2023.

[82] Z. Zhong and R. Buyya, "A cost-efficient container orchestration strategy in kubernetes-based cloud computing infrastructures with heterogeneous resources," *ACM Trans. Internet Technol.*, vol. 20, no. 2, pp. 1–24, May 2020.

[83] M. Goudarzi, H. Wu, M. Palaniswami, and R. Buyya, "An application placement technique for concurrent IoT applications in edge and fog computing environments," *IEEE Trans. on Mobile Computing*, vol. 20, no. 4, pp. 1298–1311, 2020.

[84] K. S. Gill, A. Sharma, and S. Saxena, "A systematic review on game-theoretic models and different types of security requirements in cloud environment: challenges and opportunities,"

Archives of Computational Methods in Engineering, pp. 1–34, 2024. doi: 10.1007/s11831-024-10095-6

[85] S. Filiposka, A. Mishev, and K. Gilly, "Mobile-aware dynamic resource management for edge computing," *Trans. Emerging Tel. Tech.*, vol. 30, no. 6, Jun. 2019. doi: 10.1002/ett.3626

[86] A. Mishra and A. K. Ray, "Multi-access edge computing assisted ultra-low energy scheduling and harvesting in multi-hop wireless sensor and actuator network for energy neutral self-sustainable next-gen cyber-physical system," *Future Generation Computer Systems*, vol. 141, 2023. doi: 10.1016/j.future.2022.11.023

[87] C. Lin, Y. Li, M. Ahmed, and C. Song, "Piece-wise pricing optimization with computation resource constraints for parked vehicle edge computing," *Peer-to-Peer Networking and Applications*, vol. 16, no. 2, pp. 709–726, 2023.

[88] I. C. Kanupriya and R. K. Goyal, "Computation offloading techniques in edge computing: A systematic review based on energy, QoS and authentication," *Concurrency and Computation*, vol. 36, no. 13, p. e8050, Jun. 2024. doi: 10.1002/cpe.8050

[89] E. F. Coutinho, F. R. de Carvalho Sousa, P. A. L. Rego *et al.*, "Elasticity in cloud computing: A survey," *Annals of Telecommunications-Annales des Télécommunications*, vol. 70, pp. 289–309, 2015. doi: 10.1007/s12243-014-0450-7

[90] M.-N. Tran and Y. Kim, "Optimized resource usage with hybrid auto-scaling system for knative serverless edge computing," *Future Generation Computer Systems*, vol. 152, pp. 304–316, 2024. doi: 10.1016/j.future.2023.11.010

[91] Z. Shojaee rad, M. Ghobaei-Arani, and R. Ahsan, "Memory orchestration mechanisms in serverless computing: a taxonomy, review and future directions," *Cluster Computing*, pp. 1–27, 2024. doi: 10.1007/s10586-023-04251-z

[92] S. S. Gill, I. Chana, M. Singh, and R. Buyya, "RADAR: Self-configuring and self-healing in resource management for enhancing quality of cloud services," *Concurrency and Computation*, vol. 31, no. 1, Jan. 2019. doi: 10.1002/cpe.4834

[93] R. Morabito, V. Cozzolino, A. Y. Ding, N. Beijar, and J. Ott, "Consolidate IoT edge computing with lightweight virtualization," *IEEE network*, vol. 32, no. 1, pp. 102–111, 2018.

[94] O. Ascigil, A. G. Tasiopoulos, T. K. Phan, V. Sourlas, I. Psaras, and G. Pavlou, "Resource provisioning and allocation in function-as-a-service edge-clouds," *IEEE Trans. on Services Computing*, vol. 15, no. 4, pp. 2410–2424, 2021.

[95] J. Kumar, A. Rani, and S. K. Dhurandher, "Convergence of user and service provider perspectives in mobile cloud computing environment: Taxonomy and challenges," *Int. J. Communication*, vol. 33, no. 18, Dec. 2020. doi: 10.1002/dac.4636

[96] N. Chaurasia, M. Kumar, R. Chaudhry, and O. P. Verma, "Comprehensive survey on energy-aware server consolidation techniques in cloud computing," *The Journal of Supercomputing*, vol. 77, 2021. doi: 10.1007/s11227-021-03760-1

[97] G. Portella, G. N. Rodrigues, E. Nakano, and A. C. M. A. Melo, "Statistical analysis of Amazon EC2 cloud pricing models," *Concurrency and Computation*, vol. 31, no. 18, Sep. 2019. doi: 10.1002/cpe.4451

[98] A. Gharaibeh, A. Khreishah, M. Mohammadi, A. Al-Fuqaha, I. Khalil, and A. Rayes, "Online auction of cloud resources in support of the internet of things," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1583–1596, 2017.

[99] S. Song, S. Ma, L. Yang, J. Zhao, F. Yang, and L. Zhai, "Delay-sensitive tasks offloading in multi-access edge computing," *Expert Systems with Applications*, vol. 198, p. 116730, 2022. doi: 10.1016/j.eswa.2022.116730

[100] M. Adhikari, M. Mukherjee, and S. N. Srirama, "DPTO: A deadline and priority-aware task offloading in fog computing framework leveraging multilevel feedback queueing," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 5773–5782, 2019.

[101] M. Gamsiz and A. H. Özer, "An energy-aware combinatorial virtual machine allocation and placement model for green cloud computing," *IEEE Access*, vol. 9, pp. 18625–18648, 2021. doi: 10.1109/ACCESS.2021.3054559

[102] C. Li, J. Liu, B. Lu, and Y. Luo, "Cost-aware automatic scaling and workload-aware replica management for edge-cloud environment," *Journal of Network and Computer Applications*, vol. 180, 103017, 2021. doi: 10.1016/j.jnca.2021.103017

Nisha Saini currently immersed in her Ph.D. study in computer science and engineering at Deenbandhu Chhotu Ram University of Science and Technology, Murthal, Sonipat, Haryana, India. She holds academic distinctions with a B.Tech. degree in information technology and an M.Tech. degree in computer science and engineering, both conferred by the same University. She possessed a four-year tenure of teaching experience prior to her current academic pursuits. Her research mainly focuses on the Internet of Things (IoT), cloud computing, and edge computing.

Jitender Kumar received the M. Tech. degree in computer science and engineering from the Guru Jambheshwar University of Science and Technology, Hisar, Haryana, India, in 2008. He has completed his Ph.D. study at Deenbandhu Chhotu Ram University of Science and Technology, Murthal, Sonipat, Haryana, India. He is an assistant professor at the Department of Computer Science and Engineering, Deenbandhu Chhotu Ram University of Science and Technology. His research interests include cloud computing and mobile computing.