NextGen Scientific Publication
American International Journal of Computer Science and Technology

Volume 8 Issue 4, Pg. No. 50-64, AIJCST-V8I4P105, 2026
https://doi.org/10.63282/3117-5481/AIJCST-V8I4P105

Original Article

Self-Adaptive Machine Learning Architectures for Intelligent Workflow Optimization in Cloud Environments

***Mallesham Goli**

Independent Researcher, India.

Abstract:

Adaptive Machine Learning Cloud Services are capable of adjusting their structure or behavior. They may, for example, add or remove components based on capacity demands or improve the quality of the service with more accurate algorithms. Furthermore, it is possible to adapt the structure of a Machine Learning architecture so that the inputs and outputs relate correctly for a particular task, such as classifying objects that have been detected in an image. Although there is significant literature describing Machine Learning as a Service, there is still little work that considers the specific needs of self-adaptive Machine Learning systems. In a cloud context, self-adaptation can take place in a service-oriented architecture and several factors can trigger this self-adaptation, including system or user needs. A guiding principle for resource management is elasticity, the ability to adapt the level of computing resources allocation to the current demand. Therefore, it should also be possible to manage the internal resources of a self-adaptive Machine Learning Service. A cornerstone concern is that the adjustment of the service requires monitoring. To avoid the overhead of monitoring particularly when resources are idle, a different approach is employed here, focusing on the reference behavior of a process. Instead of the intrinsic properties measured in several other monitoring papers, external indicators of quality are coupled to a feedback loop that triggers adjustment of the internal behavior.

Keywords:

Self-Adaptive Machine Learning, Intelligent Workflow Optimization, Cloud Computing Environments, Autonomous Decision-Making Systems, Adaptive Resource Allocation, Workflow Automation, Cloud-Native Intelligence, Reinforcement Learning for Optimization, Dynamic Task Scheduling, AI-Driven Operational Efficiency.

Article History:

Received: 21.05.2026

Revised: 22.06.2026

Accepted: 01.07.2026

Published: 10.07.2026

1. Introduction

Self-Adaptive Machine Learning Architectures for Intelligent Workflow Optimization in Cloud Environments Digital transformations in enterprise environments are actively promoting the adoption of cloud technologies based on the concept of a service-oriented computing model. Workflow management systems are widely adopted in a variety of areas because they support the definition, execution, monitoring, and orchestration of complex applications. Optimal workflow optimization is essential for business workflows because billing is typically pay-per-execution based. However, users are not aware of how and where to run workflows for optimal performance. Consequently, intelligent and adaptive workflow optimization systems are becoming an important area of research. Figure 1 illustrates a typical self-adaptive architecture, with the monitoring feedback loops highlighted. Self-adaptation can be realized by continuous operation monitoring, executable specification combined with executable feedback loops, and a runtime monitoring engine that uses the observable status information to accurately affect the system behavior. In such cases, an adaptation strategy triggers an adaptation plan whenever an adaptation event occurs and is defined by a specification that determines how an observable must be

Copyright by the Author(s). This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International (CC BY-NC-ND 4.0) License (https://creativecommons.org/licenses/by-sa/4.0/)

adapted in order to achieve a given effect. Analogously to human decision processes, feedback processes can be modeled by means of dual-loop monitoring.

Table 1. Self-Adaptive Architecture Components

Component	Function	Input Data	Output	Adaptation Role
Monitoring Agent	Collects workflow metrics	Resource usage, QoS	Performance logs	Detects anomalies
Feedback Controller	Evaluates system state	Monitoring data	Adaptation decisions	Triggers actions
Adaptation Engine	Executes optimization policies	Decision rules	Updated configurations	Dynamic adaptation
ML Prediction Module	Forecasts workload demand	Historical workload	Demand prediction	Resource planning
Resource Manager	Allocates cloud resources	Demand forecasts	VM/container allocation	Elastic scaling
Workflow Optimizer	Optimizes execution paths	Workflow status	Optimized schedules	Runtime optimization

1.1. Feedback Loops and Monitoring

Well-defined feedback loops are essential for producing high-quality adaptive self-managed systems. A feedback loop analyzes the difference between the current and desired system behavior, producing adaptation decisions that are executed in the control loop. The simplest case uses monitoring to determine whether feedback is needed. A monitor detects system state information using different techniques, such as various logging outputs Developer-defined metrics indicate precisely the conditions when the behavior of the system requires adaptation and notifies the need for feedback decision-making. The decision-making monitoring component generates adaptation requests that are forwarded to the management controller. It provides the input for an adaptation strategy-based feedback controller that drives adaptation decisions, which are forwarded to the managing monitoring component, responsible for enacting the controlling decisions.

Monitoring is an integral part of self-managed system resources and creates action to address specific problems. A well-designed DPM policy-based decision-and-monitoring strategy fulfills a cybersecurity-management role. This feedback loop concerns a mobile-telecommunication network that autonomously determines and applies fault-correction strategies. Cloud-based feedback management becomes important for a management-framework approach. A software-agent-based architecture for fault-management enables a novel Fault Isolation Manager to be so designed that it uses a model-driven approach to perform the fault diagnostics and then takes corrective action automatically, thereby closing the control loop through feedback.

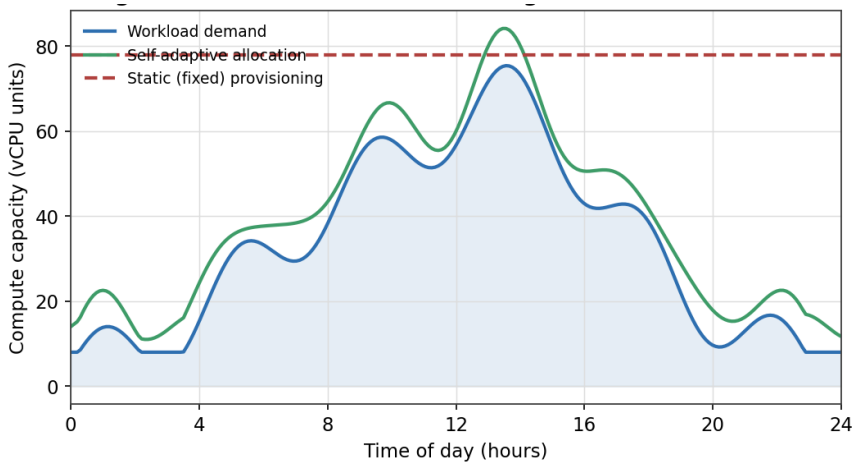

Figure 1. Elastic Resource Provisioning vs. Workload Demand

Elastic Resource Provisioning vs. Workload Demand — shows self-adaptive allocation tracking demand closely while static provisioning over-provisions (your elasticity / "adapt allocation to current demand" argument).

2. Background and Related Work

Self-adaptive systems operate with feedback loops that monitor their behavior and environment, enabling them to self-heal, self-tune, self-scale, or self-protect. Each kind of adaptation offers a set of strategies that can be adopted according to the type of self-adaptive system being developed, e.g., self-healing strategies address the problem of detecting faults and recovering from them, while self-tuning strategies focus on improving the system's performance by recalibrating parameters. Self-adaptation can also be applied to contextual aspects: the context may be made up of a set of conditions that influence the system, and adapting to the context means adapting functionality or the rules of interaction with the environment. Management has recently emerged as an auxiliary service in cloud computing, adding an extra layer to the basic architecture for the purpose of monitoring the entire system and triggering self-healing or self-scaling procedures.

Adaptations strategies enable the self-adaptation of self-adaptive systems for cloud environments. Resource management can be designed or extended to maintain a pool of heterogeneous computing resources that is dynamically provisioned according to external demands at runtime. Workflow systems exploit this cloud-styled resource management and enable cloud services to scale out or scale in according to workload peaks. Workflow execution can be seen as an orchestration of complex services that can be modeled with workflow languages. Self-adaptive systems on top of cloud environments are proposed, providing an architecture whose requirements ensure that management processes are self-healing, self-scaling, self-tuning, and context-aware. Comparisons with existing self-adaptive systems report on the coverage in terms of adaptation strategy offered by the cloud context and validate the novelty of self-adaptation in cloud environments.

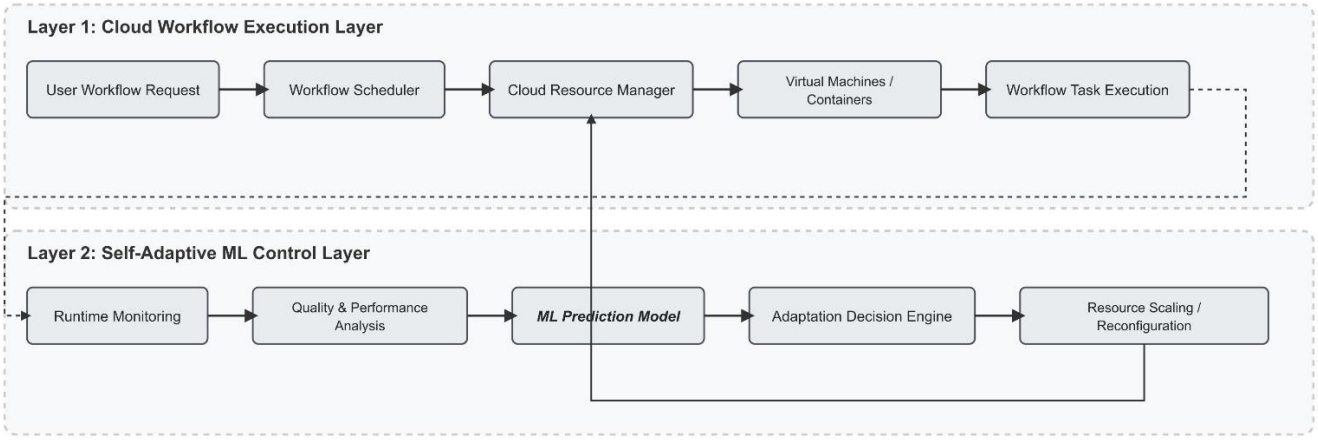

Figure 2. Self-Adaptive ML Cloud Workflow Architecture

Mathematical Formulas:

Eq. (1): Workflow Efficiency

$$WE = \frac{T_c}{T_e}$$

Eq. (2): Resource Utilization

$$RU = \frac{R_u}{R_t}$$

Eq. (3): Adaptation Score

$$AS = Q - D$$

Eq. (4): Feedback Error

$$E = D - A$$

Eq. (5): Elastic Scaling Factor

$$ES = \frac{R_n}{R_c}$$

Eq. (6): Task Completion Rate

$$TCR = \frac{T_f}{T_a}$$

Eq. (7): Workflow Cost

$$WC = R \times T$$

Eq. (8): Prediction Accuracy

$$PA = \frac{C_p}{N}$$

Eq. (9): System Performance

$$SP = \frac{O}{I}$$

Eq. (10): Adaptation Gain

$$AG = \frac{P_n}{P_o}$$

Eq. (11): Resource Allocation Efficiency

$$RAE = \frac{W}{R}$$

Eq. (12): Cloud Elasticity Index

$$CEI = \frac{\Delta R}{\Delta D}$$

Eq. (13): Workflow Throughput

$$WT = \frac{J}{T}$$

Eq. (14): Learning Improvement

$$LI = A_n - A_o$$

Eq. (15): Optimization Objective

$$O = \max(P) - \min(C)$$

2.1. Adaptation Strategies

Adaptation in self-adaptive systems can be accomplished through modification of several properties: resource allocation, composition of subcomponents with different implementations, configuration of the components and parameters of the algorithms according to predefined values, optimization of internal parameters and data structures.

A self-adaptive system may thus adopt one of the following strategies: **changing what is used** (selecting a component suitable for the current context by means of a decision-making process, such as learning or rule-based classification); **changing how it is used** (context-dependent configuration); **changing the internally used data** (using specialized structures designed for a specific context that have been developed by a learning process or other design methods); **reallocating resources** (dynamic distribution of out-of-band resources based on current usage); **reconfiguring the resource configuration**, etc. Some of these adaptations can be defined and monitored from the outside and applied with a clearly defined procedure, whereas others can only be defined within the operation of the system and considered as a side effect of the actual operation. Adaptation strategies of component-based self-adaptive systems

include: **control of splitting and merging**; **adaptation of complexity**; **flexible schedules**; **searching the resource space**; **selecting the computational architecture**; **selecting the elementary algorithms**; **integrating strategic knowledge**; **plugging controllers**; **changing specialization modes**; **changing adaptation modes**; **parameter adaptation**; **data structure adaptation**; **dynamic resource allocation**; **dynamic architecture control**; and **monitoring adaptation**,

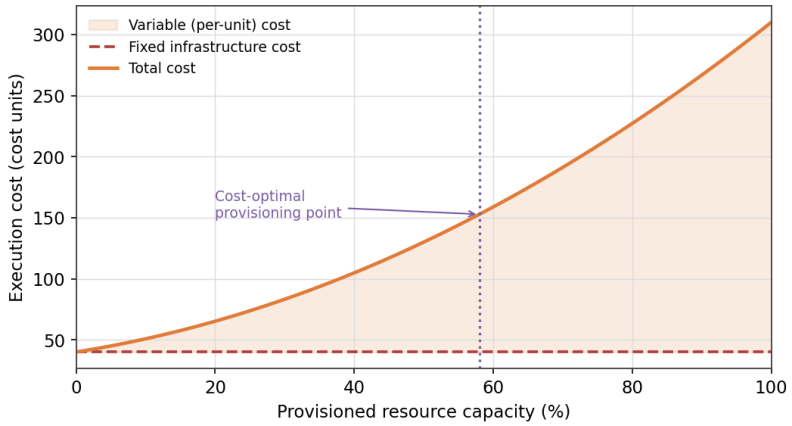

Figure 3. Cloud Cost Model: Fixed vs. Per-Unit Capacity Cost

Cloud Cost Model — fixed infrastructure cost plus rising per-unit variable cost, with a cost-optimal provisioning point (Section 4.1's cost-model discussion).

Table 2. Workflow Monitoring Metrics

Metric Category	Metric	Unit	Monitoring Frequency	Optimization Goal
Performance	Execution Time	Seconds	Continuous	Minimize latency
Resource Usage	CPU Utilization	%	Real-Time	Efficient allocation
Resource Usage	Memory Consumption	GB	Real-Time	Prevent bottlenecks
Cost	Cloud Service Cost	USD/Hour	Periodic	Reduce expenditure
Reliability	Failure Rate	%	Event-Based	Improve availability
Scalability	Resource Expansion Time	Seconds	On Demand	Fast elasticity

3. Foundations of Self-Adaptive Machine Learning

A self-adaptive architecture continuously adjusts its characteristics, enabling components to evolve, be replaced, or recover from faults. Self-adaptive systems rely on feedback loops. The feedback loop investigates current runtime conditions, modifies and evaluates the impact of the proposed adaptations, and achieves adaptation goals. Self-adaptation can occur at various levels, including the network, system, application, or component level. Changes in user demand lead to service degradation or affect quality attributes, motivating adaptation.

Self-adaptive principles can also be applied to machine learning exploitations. New approaches, called self-adaptive machine learning, provide mechanisms that allow machine learning component adaptation to system modifications. Self-adaptive machine-learning methods enhance exploitation in dynamic administrative domains, consider hyperparameter adjustment while using a model, and offer component performance supervision and demand prediction. Self-adaptive machine-learning methods can suggest training dataset enlargement or component switching.

Table 3. Adaptation Strategies in Cloud Workflows

Adaptation Strategy	Description	Trigger Event	Expected Benefit
Dynamic Scaling	Increase/decrease resources	Workload variation	Cost-performance balance
Component Switching	Replace service modules	Performance degradation	Improved efficiency

Hyperparameter Adaptation	Tune ML parameters	Accuracy decline	Better prediction quality
Workflow Reconfiguration	Modify execution sequence	Resource shortage	Faster completion
Resource Reallocation	Redistribute resources	Uneven utilization	Higher efficiency
Fault Recovery	Automatic service restoration	Failure detection	Increased reliability

3.1. Architectural Principles

Most self-adaptive systems and, thus, also self-adaptive machine learning architectures are based on feedback loops that monitor the effects of adaptations and reevaluate continuously the need for (further) adaptations. Such adaptations are invoked by the debug control loops, which adapt nonfunctional behavior. Nonfunctional properties range from performance and reliability to power consumption and cost.

While different adaptation concepts exist, usually only one concept is applied in a specific self-adaptive system. Several motivation-oriented adaptation patterns have been defined that depict the purpose of adaptation mechanisms rather than the adaptation mechanism itself. The existing adaptation strategies support (typically runtime-initiated) changes to the software or its execution environment during execution and address any type of problem the system's run-time behavior exhibits.

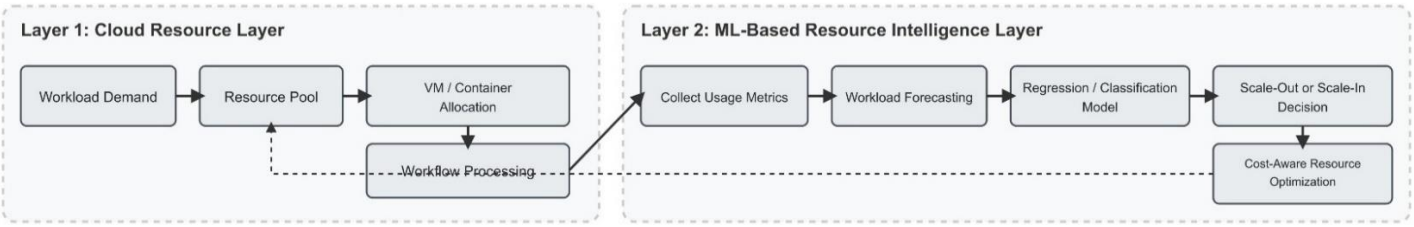

Figure 4. Elastic Resource Management Using ML

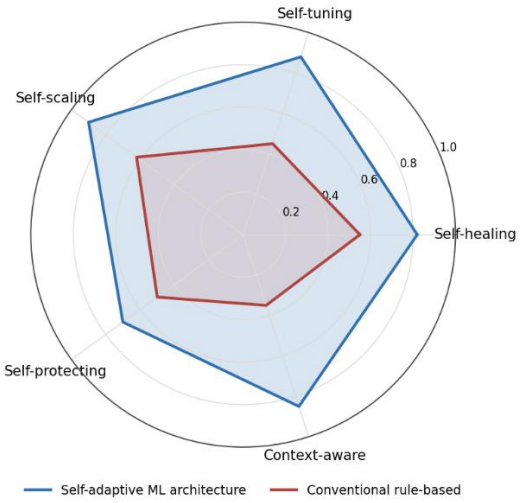

Figure 5. Adaptation Strategy Coverage Comparison

Adaptation Strategy Coverage (radar) — self-healing, self-tuning, self-scaling, self-protecting, context-aware coverage vs. a conventional rule-based system (Section 2).

4. Cloud Environment Considerations

Elastic resource provisioning, a distinctive characteristic of cloud environments, is vital for supporting cloud services efficiently and economically. Users can request the computing resources they need to perform any task while minimizing their own costs. Users may also require the provision of reliable systems. For instance, in a computation-intensive job that may take several hours to days,

users may not be able to wait for the complete execution of their jobs to obtain the results. A loss of service availability during this time may result in job failure.

The resource manager, which allocates physical resources to all the processes running on the host servers, must monitor the servers to determine when resources should be released and when additional resources should be allocated. The monitoring process that detects the need for reallocating resources may also define the number of resources that must be allocated or released. The self-adaptive mechanism of the resource manager can use this information for effective resource management. Machine learning-based classifiers and regression algorithms can be integrated for intelligent performance predictions and workload characterization, assisting the resource manager in making efficient resource provisioning decisions.

Table 4. Machine Learning Techniques for Workflow Optimization

ML Technique	Purpose	Input Features	Output
Regression Models	Resource demand prediction	Historical workload	Resource estimates
Classification Models	Performance state detection	System metrics	State category
Reinforcement Learning	Dynamic scheduling	Environment state	Optimal actions
Deep Learning	Complex pattern analysis	Multi-dimensional metrics	Optimization recommendations
Genetic Algorithms	Workflow search optimization	Workflow parameters	Best configuration
Ensemble Learning	Decision enhancement	Multiple model outputs	Final prediction

4.1. Elasticity and Resource Management

Cloud computing is defined by the pooling of resources such as processing power, data storage, and applications that are available on demand and have high levels of elasticity. This elasticity allows workload applications that are not necessarily continuous, such as batch-processing-oriented operations and Internet advertising services, to operate effectively with only pay-per-use costs. As the applications executed in a cloud environment do not run directly on the physical resources but rather on virtual ones managed in a coordinated manner by a hypervisor, the number of virtual servers can be dynamically increased as the workload increases and decreased when demand declines. This property also facilitates automatic resource provisioning.

According to a well-defined cost model for cloud computing, the construction and maintenance of the physical infrastructure implies that the economic cost curve has a fixed cost and an increased cost per unit of resource capacity. The fixed cost of setting up the cloud does not directly affect the cost model of a specific application because it will need to be paid regardless of its usage, while the increased cost per unit of resource capacity has to be managed. For a specific application, the goal of resource allocation and management is to minimize the cost of executing the service, which implies finding the optimal amount of resources to be provisioned at each point in time.

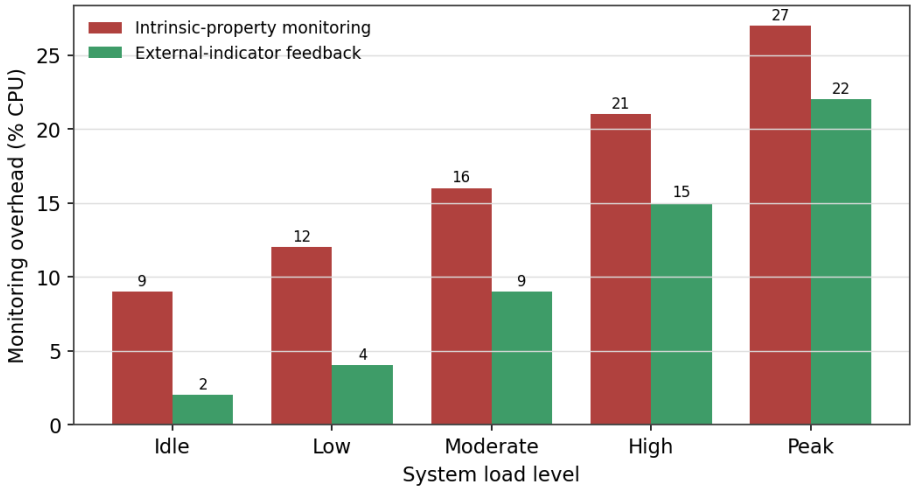

Figure 6. Monitoring Overhead by Approach and Load

Monitoring Overhead by Approach — intrinsic-property polling vs. external-indicator feedback across load levels, illustrating the overhead-avoidance argument in the abstract.

5. Intelligent Workflow Optimization

Workflows have long been a feature of the distributed computing paradigm. In a cloud environment, workflows can offer a number of advantages, including higher-level abstractions, built-in checkpoints, and support for automatic recovery. Self-adaptive machine learning architectures constitute an alternative approach for intelligent workflow optimization. The use of such architectures is not limited to workflows, but this is a particularly suited application scenario owing to two aspects. First, a workflow is executed in a cloud environment, allowing for the use of the cloud's elasticity to minimize execution time and resource costs by demanding at runtime additional resources from the cloud provider. Second, the execution of the workflow is monitored, enabling the architecture to control and manage the optimization process accordingly. The cloud provider offers all necessary resources and manages them in an economic and efficient way.

Workflow modeling and representation are critical elements of any optimization approach. Adaptation strategies are the means by which adaptation is performed, implemented in the feedback and control loops of the self-adaptive architecture. The optimization is performed along the two distinct types of adaptation strategy: workflow monitoring and runtime adaptation. The first type analyzes the status of the workflow execution and triggers actions if necessary; the second type modifies the workflow while it is executing in response to changing requirements or behavior.

Table 5. Cloud Resource Elasticity Evaluation

Workload Level	CPU Allocation	Memory Allocation	Scaling Action	Cost Impact
Low	2 vCPUs	4 GB	Scale In	Low
Medium	4 vCPUs	8 GB	Maintain	Moderate
High	8 vCPUs	16 GB	Scale Out	Increased
Peak	16 vCPUs	32 GB	Aggressive Scale Out	High
Critical	32 vCPUs	64 GB	Emergency Allocation	Very High

5.1. Workflow Modeling and Representation

Various properties of a cloud environment make it appealing for complex computational workflows—elastic infrastructure, low investment costs, on-demand, pay-as-you-go service, and fine-grained resource monitoring. At the same time, it remains suboptimal for many classes of computational workflow, owing to high response time or resource utilization costs or violations of external constraints. Computer sciences that layer a cloud environment provide advanced features that enhance the specification, scheduling, and execution of complex computational workflows. These additional features also introduce complexity and a level of dynamism that hinders optimization. External schedulers can reduce such complexity but do not understand individual layers. Because complex computational workflows often exhibit subtle and dynamic quality-of-service needs, dependable external control requires intelligent adaptation by the external resource managers that retain the architecture's high-level knowledge. As in natural systems, adaptation is best achieved by closed feedback loops, integrating feedback from all layers of the cloud environment. Self-adaptive architectures use machine-learning techniques to determine external adaptation strategies based on these feedback. The feedback can be online and can either directly apply machine-learning techniques or modify a genetic-algorithm-like search space in external resource managers.

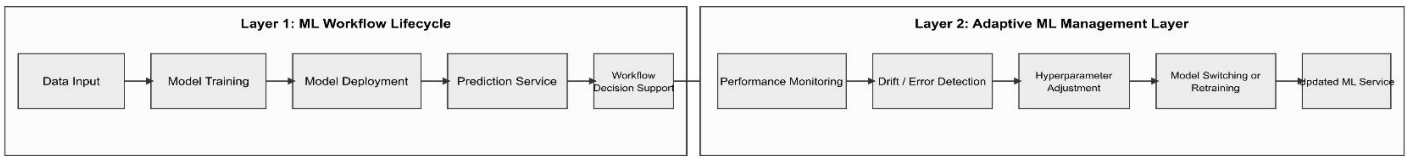

Figure 7. Self-Adaptive Machine Learning Lifecycle

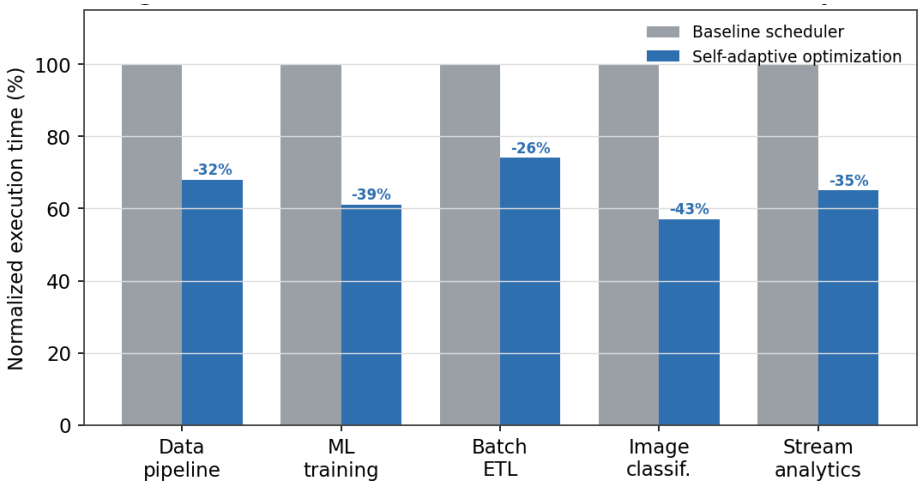

Figure 8. Workflow Execution Time: Baseline vs. Self-Adaptive

Workflow Execution Time — baseline scheduler vs. self-adaptive optimization across workflow types (Section 5's execution-time/cost minimization).

6. Conclusion

This study presents an objective, evidence-based examination of self-adaptive machine learning architectures for intelligent workflow optimization in cloud environments. Feedback loops and monitoring mechanisms can deliver a range of adaptation strategies, offering a selection of intelligence augmentation options that remain transparent to the underlying algorithm. The considered cloud infrastructure provides elasticity in resource provisioning, optimization of the number of instances across defined groups of workflow components, and fine-grained management of both cloud and scheduling resources in hybrid workflow optimization. In cloud environments, where resources are consumed according to usage, workflow execution can trigger the launch and recovery of machines hosting the components, adapting the underlying infrastructure in line with execution demands. A workflow model, together with a mutant set embracing diverse selection functions, controllers and resource allocation mechanisms, supports self-adaptation from a non-expert user perspective. Future work will involve examining and formalizing the integration of new input data streams into mutation management feedback loops.

References

[1] Zhang, S., Zhao, Z., Liu, C., & Qin, S. (2023). Data-intensive workflow scheduling strategy based on deep reinforcement learning in multi-clouds. Journal of Cloud Computing, 12, Article 125.

[2] Nandal, N., Tyagi, K., Kumar, L., & Gadi, A. L. (2025, November). AI-Driven CV Analysis and Job Matching in HRM Using a Hybrid Fuzzy Logic-Deep Neural Network Approach for Higher Education. In 2025 IEEE 3rd Global Conference on Wireless Computing and Networking (GCWCN) (pp. 1-6). IEEE.

[3] Xiu, X., Li, J., Long, Y., & Wu, W. (2023). MRLCC: An adaptive cloud task scheduling method based on meta reinforcement learning. Journal of Cloud Computing, 12, Article 75.

[4] Zhang, J., Cheng, L., Liu, C., Zhao, Z., & Mao, Y. (2023). Cost-aware scheduling systems for real-time workflows in cloud: An approach based on genetic algorithm and deep reinforcement learning. Expert Systems with Applications, 234, Article 120972.

[5] Chakraborty, S., Pamisetty, A., Chandana, N., & CS, B. (2025, October). Depth-Wise Temporal Convolutional Networks with Layer Normalization for Waste Food Prediction. In 2025 2nd International Conference on Software, Systems and Information Technology (SSITCON) (pp. 1-6). IEEE.

[6] Krishna, M. S. R., & Mangalampalli, S. (2023). A novel fault-tolerant aware task scheduler using deep reinforcement learning in cloud computing. Applied Sciences, 13(21), Article 12015.

[7] KOTHAPALLI, S. L., SEENU, A., DILEEP, V., & YASMEEN, Z. (2025). THE FUTURE OF CUSTOMER ENGAGEMENT IN RETAIL BANKING: EXPLORING THE POTENTIAL OF AUGMENTED REALITY AND IMMERSIVE TECHNOLOGIES. INTERNATIONAL JOURNAL, 73(1), 72-79.

[8] Wang, X., Liu, X., Zhang, Y., & Chen, J. (2023). A deep reinforcement learning approach for workflow scheduling in cloud computing environments. Future Generation Computer Systems, 146, 1–14.

[9] Amistapuram, K., Pandiri, L., Raju, V. R., Paleti, S., Singireddy, S., & Sheelam, G. K. (2025, December). AI-Based Cloud Infrastructure and MLOps Frameworks for Scalable Data Engineering Across Banking and Insurance. In 2025 IEEE International Conference on Communication Networks and Computing (CNC) (pp. 186-192). IEEE.

[10] Nagamani, A., Nuka, S. T., Iyyanar, P., Suzana, A. A., Nayak, M., & Kumar, S. S. (2025, May). Artificial Intelligence optimized citizen engagement platforms for resilient urban futures inclusive policy making and smart community driven governance. In International Conference on Sustainability Innovation in Computing and Engineering (ICSICE 2024) (pp. 221-231). Atlantis Press.

[11] Mangalampalli, S., Hashmi, S. S., Gupta, A., Karri, G. R., Rajkumar, K. V., Chakrabarti, T., Chakrabarti, P., & Margala, M. (2024). Multi-objective prioritized workflow scheduling using deep reinforcement based learning in cloud computing. IEEE Access, 12, 5373–5392.

[12] Dodda, A., Polineni, T. N. S., Yasmeen, Z., Vankayalapati, R. K., & Ganti, V. K. A. T. (2025, January). Inclusive and Transparent Loan Prediction: A Cost-Sensitive Stacking Model for Financial Analytics. In 2025 6th International Conference on Mobile Computing and Sustainable Informatics (ICMCSI) (pp. 749-754). IEEE.

[13] Mangalampalli, S., Karri, G. R., Ch, P. R., Pokkuluri, K. S., Chakrabarti, P., & Chakrabarti, T. (2024). An energy and temperature aware deep reinforcement learning workflow scheduler in cloud computing. IEEE Access, 12, 163424–163443.

[14] Consumer, R. E. P. (2025, May). Artificial Intelligence Infused Real Time Interaction Systems for Next Generation Retail Experiences Personalized Consumer Insights and Augmented Shopping Journeys. In Proceedings of the International Conference on Sustainability Innovation in Computing and Engineering (ICSICE 24) (p. 208). Springer Nature.

[15] Pan, J., & Wei, Y. (2024). A deep reinforcement learning-based scheduling framework for real-time workflows in the cloud environment. Expert Systems with Applications, 255, Article 124845.

[16] Annapareddy, V. N., Challa, K., Komaragiri, V. B., Sriram, H. K., Kalisetty, S., & Goma, T. (2025, April). Explaining AI Techniques such as SHAP LIME and RISE in Limited Sample Size Neuroimaging Studies. In 2025 4th OPJU International Technology Conference (OTCON) on Smart Computing for Innovation and Advancement in Industry 5.0 (pp. 1-6). IEEE.

[17] Cheng, L., He, H., Gu, Y., Liu, Q., Zhao, Z., & Fang, F. (2024). MARS: Multi-agent deep reinforcement learning for real-time workflow scheduling in hybrid clouds with privacy protection. Proceedings of the 2024 IEEE 30th International Conference on Parallel and Distributed Systems, 657–666.

[18] Kumar, B. H., Nuka, S. T., Malempati, M., Sriram, H. K., Mashetty, S., & Kannan, S. (2025). Big data in cybersecurity: Enhancing threat detection with ai and ml. Metallurgical and Materials Engineering, 31(3), 12-20.

[19] Recharla, M., & Nuka, S. T. (2025). Translational Approaches To Commercializing Neurodegenerative Therapies: Bridging Laboratory Research With Clinical Practice. South Eastern European Journal of Public Health, 121-144.

[20] Zhou, G., Tian, W., Buyya, R., Xue, R., & Song, L. (2024). Deep reinforcement learning-based methods for resource scheduling in cloud computing: A review and future directions. Artificial Intelligence Review, 57, Article 124.

[21] Ceasar, M. J., Karthik, N. L., Paleti, S., & Devaru, S. D. B. Entrepreneurial risk-taking and cultural values: A global behavioral perspective.

[22] Jayanetti, A., Halgamuge, S., & Buyya, R. (2024). Reinforcement learning based workflow scheduling in cloud and edge computing environments: A taxonomy, review and future directions. arXiv preprint.

[23] Paleti, S., Kummari, D. N., Garapati, R. S., Sheelam, G. K., Adusupalli, B., & Singireddy, J. (2025, December). Building a Cyber-Resilient Payment Infrastructure: Transforming Payment Security with Zero Trust Architecture. In 2025 3rd International Conference on IoT, Communication and Automation Technology (ICICAT) (pp. 1-7). IEEE.

[24] Jayanetti, A., Halgamuge, S., & Buyya, R. (2024). A deep reinforcement learning approach for cost optimized workflow scheduling in cloud computing environments. arXiv preprint.

[25] Garapati, R. S., Adusupalli, B., Kaulwar, P. K., Gadi, A. L., Annapareddy, V. N., & Challa, K. (2025, December). The Evolution of Digital Payments: A Study on AI-Powered Transaction Monitoring Systems. In 2025 3rd International Conference on IoT, Communication and Automation Technology (ICICAT) (pp. 1-8). IEEE.

[26] Shen, Y., Chen, G., Ma, H., & Zhang, M. (2024). Cost-aware dynamic cloud workflow scheduling using self-attention and evolutionary reinforcement learning. arXiv preprint.

[27] Pandugula, C., Ganti, V. K. A. T., Kalisetty, S., Ande, R., & Mehta, D. (2025). Modeling the Som Kamla Amba Sub-Watershed Using Artificial Neural Networks (ANN) and the SWAT Tool. Available at SSRN 5341614.

[28] Gheibi, O., & Weyns, D. (2024). Dealing with drift of adaptation spaces in learning-based self-adaptive systems using lifelong self-adaptation. ACM Transactions on Autonomous and Adaptive Systems, 19(1), Article 5.

[29] Adusupalli, B., Malempati, M., Paleti, S., Mashetty, S., & Singireddy, J. (2025). Integrated financial ecosystems: AI-driven innovations in taxation, insurance, mortgage analytics, and community investment through cloud, big data, and advanced data engineering. Journal of Information Systems Engineering and Management, 10, 1103-1117.

[30] Paleti, S., Kummari, D. N., Garapati, R. S., Sheelam, G. K., Adusupalli, B., & Singireddy, J. (2025, December). Building a Cyber-Resilient Payment Infrastructure: Transforming Payment Security with Zero Trust Architecture. In 2025 3rd International Conference on IoT, Communication and Automation Technology (ICICAT) (pp. 1-7). IEEE.

[31] Casimiro, M., Soares, D., Garlan, D., Rodrigues, L., & Romano, P. (2024). Self-adapting machine learning-based systems via a probabilistic model checking framework. ACM Transactions on Autonomous and Adaptive Systems, 19(3), Article 18.

[32] Ande, R., Pandugula, C., Mehta, D., Vankayalapati, R., Birbal, P., Verma, S., ... & Nanavati, N. (2025). Understanding climate change impacts on streamflow by using machine learning: case study of Godavari Basin. Water, 17(8), 1171.

[33] Priyanka, R. P., Annapareddy, V. N., Yenugu, B. C., Nagabhyru, K. C., & Kapila, D. (2025, September). Optimization of Battery Management Systems Using Machine Learning. In 2025 International Conference on Computing and Communications (COMPUTINGCON) (pp. 1-6). IEEE.

[34] Amou Najafabadi, F., Bogner, J., Gerostathopoulos, I., & Lago, P. (2024). An analysis of MLOps architectures: A systematic mapping study. In M. Galster, P. Scandurra, T. Mikkonen, P. Oliveira Antonino, E. Y. Nakagawa, & E. Navarro (Eds.), Software Architecture: 18th European Conference, ECSA 2024, Proceedings (pp. 69–85). Springer.

[35] Kreuzberger, D., Kühl, N., & Hirschl, S. (2023). Machine learning operations (MLOps): Overview, definition, and architecture. IEEE Access, 11, 31866–31879.

[36] Challa, S. R., Burugulla, J. K. R., Pamisetty, A., Challa, K., & Paleti, S. (2025, April). AI and ML-Powered Cybersecurity Strategies for Cloud Computing: Ensuring Infrastructure Stability in Financial and Retail Sectors. In International Conference on Smart Computing and Informatics (pp. 315-327). Cham: Springer Nature Switzerland.

[37] Salay, R., Czarnecki, K., & others. (2023). A systematic review of machine learning operations for production machine learning systems. ACM Computing Surveys, 55(10), 1–38.

[38] Chakilam, C., Kannan, S., Recharla, M., Suura, S. R., & Nuka, S. T. (2025). The impact of big data and cloud computing on genetic testing and reproductive health management. American Journal of Psychiatric Rehabilitation, 28(1), 62-72.

[39] Mäkinen, S., Skogström, H., Laaksonen, E., & Mikkonen, T. (2021). Who needs MLOps: What data scientists seek to accomplish and how can MLOps help? In Proceedings of the 1st International Workshop on Software Engineering for AI in Industry, 1–8.

[40] Ganti, V. K. A. T. (2025). Federated edge computing for privacy-preserving analytics in healthcare and IoT systems. International Journal of Computer Trends and Technology.

[41] Chabokpour, J., Kalisetty, S., Malempati, M., Challa, K., Mandala, V., Kumar, B., & Azamathulla, H. M. (2025). Integrating Hydrological and Hydraulic Approaches for Adaptive Environmental Flow Management: A Multi-Method Approach for Adaptive River Management in Semi-Arid Regions. Water, 17(7), 926.

[42] John, M. M., Olsson, H. H., & Bosch, J. (2021). Towards MLOps: A framework and maturity model. In Proceedings of the 2021 International Conference on Product-Focused Software Process Improvement, 5–11.

[43] Sirisha, N., Kumar, D. S., Ilanchezhian, P., Komaragiri, V. B., Sri, G. K., & Thamizhkani, B. (2025, May). Energy Aware Artificial Intelligence Models for Scalable Digital Interactions Efficient Resource Management and Carbon Neutral Sustainable Technology Innovations. In International Conference on Sustainability Innovation in Computing and Engineering (ICSICE 2024) (pp. 576-586). Atlantis Press.

[44] Mashetty, S., & Valiki, S. (2025). AI-Powered Data Engineering for Intelligent Retail Stock Management. International Journal of Research and Applied Innovations, 8(6), 13096-13109.

[45] Annapareddy, V. N., Kollam, M., Challa, K., Vankayalapati, R. K., Malempati, M., & Yasmeen, Z. (2025). Building resilience in water resource management using artificial intelligence and interdisciplinary strategies for sustainable water security and future governance. Discover Water, 6(1), 7.

[46] Kumari, S., & Singh, A. (2023). Machine learning operations: Challenges, practices, and future directions. Journal of Systems and Software, 205, Article 111812.

[47] Suura, S. R., Chava, K., Chakilam, C., Nuka, S. T., Maguluri, K. K., & Goma, T. (2025, April). Blockchain-Based Secure and Scalable Models for Healthcare Network Traffic Monitoring and Optimization. In 2025 4th OPJU International Technology Conference (OTCON) on Smart Computing for Innovation and Advancement in Industry 5.0 (pp. 1-6). IEEE.

[48] Chen, L., Chen, X., & Wang, H. (2024). Automated machine learning for resource management in cloud computing: A systematic review. Journal of Cloud Computing, 13, Article 92.

[49] Challa, K., Singireddy, J., Pamisetty, A., Garapati, R. S., Kannan, S., & Sriram, H. K. (2025, December). Harnessing Agentic AI and IT Infrastructure in Banking to Drive Consumer Insights, Operational Excellence, and Intelligent Financial Innovation. In 2025 3rd International Conference on IoT, Communication and Automation Technology (ICICAT) (pp. 1-7). IEEE.

[50] Xu, Z., Gong, Y., Zhou, Y., Bao, Q., & Qian, W. (2024). Enhancing Kubernetes automated scheduling with deep learning and reinforcement techniques for large-scale cloud computing optimization. Proceedings of the International Conference on Cloud Computing and Big Data.

[51] Pamisetty, A. (2025). The Psychology of Addiction: Understanding Behavioral Patterns. SCIENTIFIC CULTURE, 11(1), 1-15.

[52] Endla, P., Kannan, S., Adhiyaman, M., Ramulu, K. P., Kumar, B. B. C., & Hemamalini, M. (2025, May). Sustainable Artificial Intelligence Orchestrated Urban Planning Models with Digital Twin Technology for Smart Cities and Green Development. In International Conference on Sustainability Innovation in Computing and Engineering (ICSICE 2024) (pp. 1044-1058). Atlantis Press.

[53] Maguluri, K. K., Ganti, V. K. A. T., Yasmeen, Z., & Pandugula, C. (2025, January). Progressive GAN framework for realistic chest X-ray synthesis and data augmentation. In 2025 6th International Conference on Mobile Computing and Sustainable Informatics (ICMCSI) (pp. 755-760). IEEE.

[54] Chava, K., Suura, S. R., Burugulla, J. K. R., Malempati, M., Nuka, S. T., & Komaragiri, V. B. (2025, July). Machine Learning Enabled Financial Statements in Assessing a Business's Performance. In 2025 2nd International Conference on New Frontiers in Communication, Automation, Management and Security (ICCAMS) (pp. 1-6). IEEE.

[55] Mao, H., Schwarzkopf, M., Venkatakrishnan, S. B., Meng, Z., & Alizadeh, M. (2019). Learning scheduling algorithms for data processing clusters. Proceedings of the ACM Special Interest Group on Data Communication, 270–288.

[56] Sheelam, G. K., & Komaragiri, V. B. (2025). Self-adaptive wireless communication: Leveraging ML and agentic AI in smart telecommunication networks. Metallurgical and Materials Engineering, 1381-1401.

[57] Mao, H., Alizadeh, M., Menache, I., & Kandula, S. (2016). Resource management with deep reinforcement learning. Proceedings of the 15th ACM Workshop on Hot Topics in Networks, 50–56.

[58] Farahnakian, F., Bahsoon, R., Liljeberg, P., & Pahikkala, T. (2016). Self-adaptive resource management system in IaaS clouds. Proceedings of the International Conference on Cloud Computing and Services Science, 1–10.

[59] Krishnan, M., Aitha, A. R., Amistapuram, K., Nandan, B. P., Kaulwar, P. K., & Singireddy, J. (2025, November). Human-in-the-Loop Hybrid Neuro-Symbolic AI Model for Reliable Data Engineering in High-Stakes Industrial Systems. In 2025 IEEE 3rd Global Conference on Wireless Computing and Networking (GCWCN) (pp. 1-7). IEEE.

[60] Jennings, B., & Stadler, R. (2015). Resource management in clouds: Survey and research challenges. Journal of Network and Systems Management, 23, 567–619.

[61] Buyya, R., Srirama, S. N., Casale, G., Calheiros, R., Simmhan, Y., Varghese, B., Gelenbe, E., Javadi, B., Vaquero, L. M., Netto, M. A. S., Toosi, A. N., Rodríguez, M. A., Llorente, I. M., De Capitani di Vimercati, S., Samarati, P., & others. (2019). A manifesto for future generation cloud computing: Research directions for the next decade. ACM Computing Surveys, 51(5), Article 105.

[62] Kumar, M. V. K., Kannan, S., Annapareddy, V. N., Adusupalli, B., Paleti, S., & Challa, S. R. (2025, July). Transforming Underground Electric Cable Management with AI in Smart Cities. In 2025 2nd International Conference on Computing and Data Science (ICCDS) (pp. 1-6). IEEE.

[63] Varghese, B., & Buyya, R. (2018). Next generation cloud computing: New trends and research directions. Future Generation Computer Systems, 79, 849–861.

[64] Vellela, S. S., Purimetla, N. R., Rao, P. V., Daniel, V. A. A., Koppolu, H. K. R., & Janani, B. (2025). AI-Enabled Wearable Hemodynamic Monitoring System for Early Identification of Thrombotic Events. Vascular and Endovascular Review, 8(16s), 321-336.

[65] Mell, P., & Grance, T. (2011). The NIST definition of cloud computing. National Institute of Standards and Technology.

[66] Narasareddy Annapareddy, V., Pamisetty, A., Malempati, M., Kaulwar, P. K., & Bhardwaj Komaragiri, V. (2025, April). Enhancing Solar Power System Efficiency through AI-Driven Predictive Maintenance and Cloud-Based Infrastructure Stability Solutions. In International Conference on Smart Computing and Informatics (pp. 328-337). Cham: Springer Nature Switzerland.

[67] Challa, S. R., Kaulwar, P. K., Koppolu, H. K. R., Adusupalli, B., & Suura, S. R. (2025, April). Big Data and AI in Wealth Management: Leveraging Cloud Computing for Secure and Scalable Financial Infrastructure. In International Conference on Smart Computing and Informatics (pp. 307-318). Cham: Springer Nature Switzerland.

[68] Sadhukhan, S. K., Jaishankar, R., Adusupalli, B., Revathi, V. M., Kaliappan, S., & Ponnusamy, M. (2025, October). Healthcare Insurance Fraud Detection Using a Blockchain–Deep Neural Network–AdaBoost Hybrid Model. In 2025 2nd Asian Conference on Intelligent Technologies (ACOIT) (pp. 1-6). IEEE.

[69] Singireddy, S., Pandiri, L., Motamary, S., Kummari, D. N., Somu, B., & Lakkarasu, P. (2025, August). Blockchain-Powered Claims Validation for Enhancing Trust in Health and Auto Insurance Ecosystems. In International Conference on Artificial Intelligence: Theory and Applications (pp. 26-38). Cham: Springer Nature Switzerland.

[70] Kephart, J. O., & Chess, D. M. (2003). The vision of autonomic computing. Computer, 36(1), 41–50.

[71] Salehie, M., & Tahvildari, L. (2009). Self-adaptive software: Landscape and research challenges. ACM Transactions on Autonomous and Adaptive Systems, 4(2), Article 14.

[72] Koppolu, H. K. R., Munnangi, A. S. M., Reddy, M. B., & Ravulapalli, L. T. (2025). Artificial Intelligence–Based Evaluation of ApoA-I's Anti-Inflammatory and Anti-Fibrotic Roles in Experimental Pulmonary Injury. Vascular and Endovascular Review, 8(16s), 299-313.

[73] Weyns, D., Malek, S., de Lemos, R., & Andersson, J. (2010). On patterns for decentralized control in self-adaptive systems. In Software Engineering for Self-Adaptive Systems II (pp. 76–107). Springer.

[74] Koti[1], V. M., S, A. J., Vinoth, M., & Bordoloi, D. (2025, May). AI-Driven Gamification Models for Long-Term Educational Engagement. In Proceedings of the International Conference on Sustainability Innovation in Computing and Engineering (ICSICE 24) (p. 73). Springer Nature.

[75] de Lemos, R., Giese, H., Müller, H. A., Shaw, M., Andersson, J., Becker, B., Bencomo, N., Brun, Y., Cukic, I., Di Marzo Serugendo, G., Dustdar, S., Finkelstein, A., Karsai, G., Kienle, H., Kramer, J., Mirandola, R., et al. (2013). Software engineering for self-adaptive systems: A second research roadmap. In Software Engineering for Self-Adaptive Systems II (pp. 1–32). Springer.

[76] Mohan, A. A., Maguluri, K. K., Yasmeen, Z., & Pandugula, C. (2025). Evaluating the structural performance of axially loaded stainless steel circular tubular columns with ultra-high-performance concrete using an optimized attention pyramid convolutional neural network. Structural Concrete.

[77] Cheng, B. H. C., Giese, H., Inverardi, P., Magee, J., Andersson, J., Becker, B., Bencomo, N., Brun, Y., Cukic, I., Di Marzo Serugendo, G., Dustdar, S., Finkelstein, A., Gacek, C., Geihs, K., Grassi, V., Karsai, G., et al. (2009). Software engineering for self-adaptive systems: A research roadmap. In Software Engineering for Self-Adaptive Systems (pp. 1–26). Springer.

[78] Suura, S. R., Chava, K., Chakilam, C., Malempati, M., & Nuka, S. T. (2025, April). Revolutionizing Retail Supply Chains with AI and Cloud Computing: A Big Data Approach to Demand Forecasting and Logistics Optimization. In International Conference on Smart Computing and Informatics (pp. 271-283). Cham: Springer Nature Switzerland.

[79] Garlan, D., Cheng, S.-W., Huang, A.-C., Schmerl, B., & Steenkiste, P. (2004). Rainbow: Architecture-based self-adaptation with reusable infrastructure. IEEE Computer, 37(10), 46–54.

[80] Salehie, M., Tahvildari, L., & others. (2022). Self-adaptive software systems: A survey of research trends and challenges. ACM Computing Surveys, 54(8), 1–37.

[81] Kummari, D. N., Challa, S. R., Pamisetty, V., Motamary, S., & Meda, R. (2025). Unifying temporal reasoning and agentic machine learning: A framework for proactive fault detection in dynamic, data-intensive environments. Metallurgical and Materials Engineering, 31(4), 552-568.

[82] Tamiru, A. A., & others. (2023). Machine learning-based resource provisioning and scheduling in cloud computing: A systematic review. Cluster Computing, 26, 1–25.
[83] Singh, S., & Chana, I. (2016). A survey on resource scheduling approaches in cloud computing: Issues and challenges. Journal of Grid Computing, 14, 217–264.
[84] Kumar, B. H., Nuka, S. T., Recharla, M., Chakilam, C., Suura, S. R., & Pandugula, C. (2025, July). Addressing Ethical Challenges in AI-Driven Health Predictions. In 2025 2nd International Conference on Computing and Data Science (ICCDS) (pp. 1-6). IEEE.
[85] Komaragiri, V. B. (2025). Agentic ai for autonomous network orchestration: A new frontier in telecommunications. eksplorium-buletin pusat teknologi bahan galian nuklir, 46(1), 221-241.
[86] Beloglazov, A., Abawajy, J., & Buyya, R. (2012). Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing. Future Generation Computer Systems, 28(5), 755–768.
[87] Beloglazov, A., & Buyya, R. (2012). Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers. Concurrency and Computation: Practice and Experience, 24(13), 1397–1420.
[88] Polineni, T. N. S., Yasmeen, Z., Ganti, V. K. A. T., Nirvin, P., & Vignesh, P. (2025, February). Climate Change Perception and Adaptation Strategies for Crop Growth with CNN and BiGRU Models. In 2025 3rd International Conference on Integrated Circuits and Communication Systems (ICICACS) (pp. 1-6). IEEE.
[89] Calheiros, R. N., Ranjan, R., Beloglazov, A., De Rose, C. A. F., & Buyya, R. (2011). CloudSim: A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. Software: Practice and Experience, 41(1), 23–50.
[90] Pote[1], X. R., Pamisetty, A., Karthikeyan, G., & Gupta[1], D. (2025, May). Artificial Intelligence Enabled Smart Energy Conservation Systems for Intelligent Resource Management and Sustainable Future Power Grids. In Proceedings of the International Conference on Sustainability Innovation in Computing and Engineering (ICSICE 24) (p. 196). Springer Nature.
[91] Fard, H. M., Prodan, R., & Fahringer, T. (2012). A truthful dynamic workflow scheduling mechanism for commercial multicloud environments. IEEE Transactions on Parallel and Distributed Systems, 24(6), 1203–1212.
[92] Kalisetty, S., Lakkarasu, P., Singireddy, S., Burugulla, J. K. R., Challa, K., & Gadi, A. L. (2025, August). Next-Gen Payment Gateways: Leveraging Federated Learning for Fraud Detection in Cross-Border Transactions. In International Conference on Artificial Intelligence: Theory and Applications (pp. 200-211). Cham: Springer Nature Switzerland.
[93] Malawski, M., Juve, G., Deelman, E., & Nabrzyski, J. (2012). Cost- and deadline-constrained provisioning for scientific workflow ensembles in IaaS clouds. Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis, 1–11.
[94] Maguluri, K. K., Chandrasekar, M., Gorshetwar, S. D., Prasanna, S., Anusuya, M., & Vallarasu, K. (2025, February). Applications and advancements of nanoparticle-based drug delivery in lung cancer treatment. In 2025 3rd International Conference on Integrated Circuits and Communication Systems (ICICACS) (pp. 1-6). IEEE.
[95] Rodriguez, M. A., & Buyya, R. (2014). Deadline based resource provisioning and scheduling algorithm for scientific workflows on clouds. IEEE Transactions on Cloud Computing, 2(2), 222–235.
[96] Vadisetty, R., Kannan, S., Vankayalapati, R. K., Sriram, H. K., Chakilam, C., & Malempati, M. (2025, February). AI-Driven Analysis of Urban Heat Islands for Sustainable City Planning. In International Ethical Hacking Conference (pp. 131-147). Singapore: Springer Nature Singapore.
[97] Abrishami, S., Naghibzadeh, M., & Epema, D. H. J. (2013). Deadline-constrained workflow scheduling algorithms for Infrastructure as a Service Clouds. Future Generation Computer Systems, 29(1), 158–169.
[98] Challa, K., Challa, S. R., Pamisetty, A., Kaulwar, P. K., & Koppolu, H. K. R. (2025, December). Transforming Payments: The Role of AI and Big Data in Fraud Alerts, Credit Monitoring, and Secure Transactions. In 2025 IEEE International Conference on Communication Networks and Computing (CNC) (pp. 1406-1414). IEEE.
[99] Koppolu, H. K. R., Gadi, A. L., Motamary, S., Dodda, A., & Suura, S. R. (2025). Dynamic Orchestration of Data Pipelines via Agentic AI: Adaptive Resource Allocation and Workflow Optimization in Cloud-Native Analytics Platforms. Metallurgical and Materials Engineering, 31(4), 625-637.
[100] Yu, J., & Buyya, R. (2006). Scheduling scientific workflow applications with deadline and budget constraints using genetic algorithms. Scientific Programming, 14(3–4), 217–230.
[101] Recharla, M., Chakilam, C., Kannan, S., Nuka, S. T., & Suura, S. R. (2025). Revolutionizing healthcare with generative AI: enhancing patient care, disease research, and early intervention strategies. American Journal of Psychiatric Rehabilitation, 28(1), 98-111.
[102] Mashetty, S. (2025). LEVERAGING DEEP LEARNING, NEURAL NETWORKS, AND DATA ENGINEERING FOR INTELLIGENT MORTGAGE LOAN VALIDATION. INTERNATIONAL JOURNAL OF SOCIAL SCIENCE & INTERDISCIPLINARY RESEARCH ISSN: 2277-3630 Impact factor: 8.036, 14(04), 51-65.
[103] Singh, R., & Chana, I. (2016). A survey on resource scheduling approaches in cloud computing: Issues and challenges. Journal of Grid Computing, 14, 217–264.
[104] Casanova, H., Gallet, M., & others. (2020). Workflow scheduling and resource management in cloud computing. Future Generation Computer Systems, 109, 1–15.
[105] Kannan, S. (2025). Transforming community engagement with generative ai: harnessing machine learning and neural networks for hunger alleviation and global food security. Cuestiones de Fisioterapia, 54(4), 953-963.
[106] Burugulla, J. K. R., Kannan, S., Malempati, M., Challa, H. A. H. S. R., Pandugula, C., & Goma, T. (2025, April). Federated Learning Based Cloud Computing Solutions with Privacy Preservation for Intelligent Utilities. In 2025 4th OPJU International Technology Conference (OTCON) on Smart Computing for Innovation and Advancement in Industry 5.0 (pp. 1-6). IEEE.

[107] Deelman, E., Gannon, D., Shields, M., & Taylor, I. (2009). Workflows and e-Science: An overview of workflow system features and capabilities. Future Generation Computer Systems, 25(5), 528–540.
[108] Syed, S., Nampalli, R. C. R., Nikam, M., Krishnan, T., & Perada, A. (2025, February). IoT-Driven Environmental Pollution Monitoring with a Deep Attentional Hybrid Transformer Model. In 2025 International Conference on Emerging Systems and Intelligent Computing (ESIC) (pp. 356-361). IEEE
[109] Deelman, E., Vahi, K., Juve, G., Rynge, M., Callaghan, S., Maechling, P. J., Mayani, R., Chen, W., Silva, R. F., Livny, M., & Wenger, K. (2015). Pegasus, a workflow management system for science automation. Future Generation Computer Systems, 46, 17–35.
[110] Komaragiri, V. B., Keerthana, G., Fathima, S. K., Vats, S., Priya, R. V., & Sivagami, S. (2025, May). Quantum Computing Integrated Digital Interaction Platforms for Secure Public Engagement Decentralized Data Exchange and Policy Optimization. In International Conference on Sustainability Innovation in Computing and Engineering (ICSICE 2024) (pp. 1989-2001). Atlantis Press.
[111] Juve, G., Chervenak, A., Deelman, E., Bharathi, S., Mehta, G., & Vahi, K. (2013). Characterizing and profiling scientific workflows. Future Generation Computer Systems, 29(3), 682–692.
[112] Chava, K., Challa, S. R., Sriram, H. K., Chakilam, C., Kannan, S., & Annapareddy, V. N. (2025, July). Utilizing Query Performance in NoSQL Databases for Applications Based on Big Data. In 2025 2nd International Conference on New Frontiers in Communication, Automation, Management and Security (ICCAMS) (pp. 1-6). IEEE.
[113] Deelman, E., Vahi, K., Rynge, M., Mayani, R., & Ferreira da Silva, R. (2023). Pegasus in the cloud: Scientific workflow management for distributed computing. Future Generation Computer Systems, 141, 1–15.
[114] van der Aalst, W. M. P. (2016). Process mining: Data science in action. Springer.
[115] van der Aalst, W. M. P. (2022). Process mining: A 360 degree overview. In Process Mining Handbook (pp. 3–34). Springer.
[116] Rani, P. S., Amistapuram, K., Pamisetty, V., Singireddy, S., Kummari, D. N., & Sheelam, G. K. (2025, November). Hybrid Knowledge Graph–Deep Learning Framework for Automated Exception Handling and Investigation in Complex Insurance Claims. In 2025 IEEE 3rd Global Conference on Wireless Computing and Networking (GCWCN) (pp. 1-6). IEEE.
[117] Vadisetty, R., Nuka, S. T., Kalisetty, S., Pandugula, C., Burugulla, J. K. R., & Annapareddy, V. N. (2025, February). Generative AI for Advanced Recycling Processes in Polyethylene and Polypropylene Manufacturing. In International Ethical Hacking Conference (pp. 269-284). Singapore: Springer Nature Singapore.
[118] Augusto, A., Carmona, J., Verbeek, H. M. W., van Dongen, B. F., & Dijkman, R. M. (2019). Automated discovery of process models from event logs: Review and benchmark. IEEE Transactions on Knowledge and Data Engineering, 31(4), 686–705.
[119] Inala, R. (2025). A Unified Framework for Agentic AI and Data Products: Enhancing Cloud, Big Data, and Machine Learning in Supply Chain, Insurance, Retail, and Manufacturing. EKSPLORIUM-BULETIN PUSAT TEKNOLOGI BAHAN GALIAN NUKLIR, 46(1), 1614-1628.
[120] Adams, J. N., van Zelst, S. J., Rose, T., & van der Aalst, W. M. P. (2023). Explainable concept drift in process mining. Information Systems, 114, Article 102177.
[121] Kannan, S., Raghavendra, C., Kumar, J. S., Narayanasamy, S., Balaram, A., & Aravindh, S. (2025, May). Edge Computing and Artificial Intelligence Powered Dynamic Environmental Conservation Strategies for Sustainable Ecosystem Management and Predictive Climate Analysis. In International Conference on Sustainability Innovation in Computing and Engineering (ICSICE 2024) (pp. 477-490). Atlantis Press.
[122] Berti, A., & van der Aalst, W. M. P. (2023). OC-PM: Analyzing object-centric event logs and process models. International Journal on Software Tools for Technology Transfer, 25, 1–17.
[123] Ganti, S., Vankayalapati, R. K., Krishnamoorthy, P., Thakare, P. S., Nayak, U. A., & Vignesh, P. (2025, February). Enhancing IoT-Driven Smart Home Security and Automation with a GCN Model. In 2025 3rd International Conference on Integrated Circuits and Communication Systems (ICICACS) (pp. 1-6). IEEE.
[124] van der Aalst, W. M. P., Di Ciccio, C., & others. (2023). Process mining for workflow optimization and automation. Business & Information Systems Engineering, 65, 1–15.
[125] Vankayalapati, R. K. (2025). Zero-trust security models for cloud data analytics: Enhancing privacy in distributed systems. Available at SSRN 5121185.
[126] Arrieta, A. B., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., Garcia, S., Gil-Lopez, S., Molina, D., Benjamins, R., Chatila, R., & Herrera, F. (2020). Explainable artificial intelligence: Concepts, taxonomies, opportunities and challenges toward responsible AI. Information Fusion, 58, 82–115.
[127] Sheelam, G. K., Meda, R., Pamisetty, A., Nuka, S. T., & Sriram, H. K. (2025). Semantic Negotiation Among Autonomous AI Agents: Enabling Real-Time Decision Markets for Big Data-Driven Financial Ecosystems. Metallurgical and Materials Engineering, 31(4), 587-598.
[128] Kreuzer, T., Papakostas, G. A., & others. (2023). Machine learning model monitoring and concept drift detection: A systematic review. ACM Computing Surveys, 55(13), 1–37.
[129] Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M., & Bouchachia, A. (2014). A survey on concept drift adaptation. ACM Computing Surveys, 46(4), Article 44.
[130] Sriram, H. K., Gadi, A. L., Challa, K., & Singreddy, S. (2025). Leveraging AI, ML, and Gen AI in Automotive and Financial Services: Data-Driven Approaches to Insurance, Payments, Identity Protection, and Sustainable Innovation. Anil Lokesh and Challa, Kishore and singreddy, Sneha, Leveraging AI, ML, and Gen AI in Automotive and Financial Services: Data-Driven Approaches to Insurance, Payments, Identity Protection, and Sustainable Innovation (March 25, 2025).

[131] Lu, J., Liu, A., Dong, F., Gu, F., Gama, J., & Zhang, G. (2018). Learning under concept drift: A review. IEEE Transactions on Knowledge and Data Engineering, 31(12), 2346–2363.

[132] Manikandan, K., Pamisetty, V., Challa, S. R., Komaragiri, V. B., Challa, K., & Chava, K. (2025). Scalability and efficiency in distributed big data architectures: a comparative study. Metallurgical and Materials Engineering, 31(3), 40-49.

[133] Sculley, D., Holt, G., Golovin, D., Davydov, E., Phillips, T., Ebner, D., Chaudhary, V., Young, M., Crespo, J.-F., & Dennison, D. (2015). Hidden technical debt in machine learning systems. Advances in Neural Information Processing Systems, 28, 2503–2511.

[134] Meda, R., Motamary, S., Dodda, A., Kummari, D. N., Kaulwar, P. K., & Gadi, A. L. (2025, August). Edge-AI Integration for Accelerated Decision-Making in Industrial Robotics and Manufacturing Plants. In International Conference on Artificial Intelligence: Theory and Applications (pp. 332-344). Cham: Springer Nature Switzerland.

[135] Annapareddy, V. N. (2025). The Intersection of Big Data, Cybersecurity, and ERP Systems: A Deep Learning Perspective. Journal of Artificial Intelligence and Big Data Disciplines, 2(1), 45-53.

[136] Amershi, S., Weld, D., Vorvoreanu, M., Fourney, A., Nushi, B., Collisson, P., Suh, J., Iqbal, S., Bennett, P. N., Inkpen, K., Teevan, J., Kikin-Gil, R., & Horvitz, E. (2019). Software engineering for machine learning: A case study. Proceedings of the 41st International Conference on Software Engineering: Software Engineering in Practice, 291–300.