










LLM-Based Adaptive Digital Twin Allocation for Microservice Workloads

Pedro Henrique Sachete Garcia¹^a, Ester de Souza Oribes¹^b, Ivan Mangini Lopes Junior¹^c,
Bráulio Marques de Souza¹^d, Angelo Nery Vieira Crestani²^e, Arthur Francisco Lorenzon³^f,
Marcelo Caggiani Luizelli¹^g, Paulo Silas Severo de Souza¹^h and Fábio Diniz Rossi²ⁱ

¹Federal University of Pampa, Brazil

²Federal Institute Farroupilha, Brazil

³Federal University of Rio Grande do Sul, Brazil

fl

Keywords: Cloud-Native Applications, Digital Twins, Large Language Models, Resource Allocation.

Abstract: Efficient resource allocation in programmable datacenters is a critical challenge due to the diverse and dynamic nature of workloads in cloud-native environments. Traditional methods often fall short in addressing the complexities of modern datacenters, such as inter-service dependencies, latency constraints, and optimal resource utilization. This paper introduces the Dynamic Intelligent Resource Allocation with Large Language Models and Digital Twins (DIRA-LDT) framework, a cutting-edge solution that combines real-time monitoring capabilities of Digital Twins with the predictive and reasoning strengths of Large Language Models (LLMs). DIRA-LDT systematically optimizes resource management by achieving high allocation accuracy, minimizing communication latency, and maximizing bandwidth utilization. By leveraging detailed real-time insights and intelligent decision-making, the framework ensures balanced resource distribution across the datacenter while meeting stringent performance requirements. Among the key results, DIRA-LDT achieves an allocation accuracy of 98.5%, an average latency reduction to 5.3 ms, and a bandwidth utilization of 82.4%, significantly outperforming heuristic-based, statistical, machine learning, and reinforcement learning approaches.


1 INTRODUCTION


The rapid proliferation of cloud-native applications (Hongyu and Anming, 2023), driven by the widespread adoption of microservices and containerized workloads, has transformed the landscape of modern data center infrastructures. These environments offer unparalleled advantages like scalability, modularity, and fault tolerance. Cloud-native applications leverage distributed architectures that allow


dynamic scaling of services based on demand, ensuring cost efficiency and resilience. Furthermore, container orchestration platforms, such as Kubernetes (Nguyen and Kim, 2022), facilitate automated deployment, scaling, and management of microservices, thereby reducing operational overhead and enabling faster application development cycles.


Despite their numerous advantages, cloud-native environments present significant challenges. The distributed nature of microservices increases the complexity of managing resources efficiently, particularly in heterogeneous data center infrastructures. Latency-sensitive applications, for instance, require precise coordination of resources to minimize delays, while workloads with fluctuating demands necessitate real-time resource allocation. Additionally, the inherent abstraction of cloud-native platforms often limits visibility into the underlying infrastructure, making it challenging to diagnose performance bottlenecks or predict failures. These limitations highlight the need


^a <https://orcid.org/0009-0005-7487-854X>


^b <https://orcid.org/0009-0001-3846-5925>


^c <https://orcid.org/0009-0006-2993-5477>


^d <https://orcid.org/0009-0002-9000-1260>

^e <https://orcid.org/0009-0003-6577-5798>

^f <https://orcid.org/0000-0002-2412-3027>

^g <https://orcid.org/0000-0003-0537-3052>

^h <https://orcid.org/0000-0003-4945-3329>

ⁱ <https://orcid.org/0000-0002-2450-1024>

for advanced tools to provide actionable insights and optimize resource management in real-time (Deng et al., 2024).

Digital Twins (DTs) (Li et al., 2021) have emerged as a promising solution to address these challenges. By creating virtual replicas of physical resources, DTs enable continuous monitoring, simulation, and optimization of networked systems. Several attempts to integrate DTs into cloud-native environments have demonstrated their potential to improve resource utilization and predict system behavior under different scenarios. However, many existing approaches rely on static rules or basic heuristics, which fail to adapt to the dynamic nature of cloud-native applications. These methods often lack the sophistication required to handle complex dependencies between microservices or to optimize performance across diverse workloads.

In recent years, there have been attempts to enhance resource allocation using machine learning models and rule-based algorithms (Buchaca et al., 2020) (Spyrou et al., 2024) (Morichetta et al., 2023). While these solutions offer some adaptability, they often struggle to scale effectively or incorporate real-time feedback from the infrastructure. Additionally, traditional approaches typically focus on isolated metrics, such as CPU or memory usage, without considering the interplay between latency, bandwidth, and workload requirements. As a result, they fail to achieve optimal performance in dynamic, heterogeneous environments.

In order to overcome these limitations, this paper introduces a novel approach that leverages Large Language Models (LLMs) (Mohammad et al., 2023), such as Llama3, to dynamically allocate Digital Twins in programmable data center environments. Our solution integrates real-time metrics into an LLM-based framework, including latency, bandwidth utilization, and workload dependencies. The LLM can make informed allocation decisions that balance resource utilization, minimize latency, and maximize system performance by processing these complex datasets. This approach addresses the limitations of previous methods and establishes a foundation for adaptive resource management in cloud-native environments.

Through comprehensive evaluation, we demonstrate our proposed solution's effectiveness compared to existing methods. The results highlight significant improvements in precision, resource utilization, and decision-making efficiency, validating the potential of LLM-powered Digital Twin allocation for modern data center infrastructures. The article is structured as follows: Section 2 provides a theoretical framework

for the technologies relevant to this research. Section 3 defines the problem addressed in this work. Section 4 reviews related works that offer solutions to this problem, highlighting their limitations and implications. Section 5 presents our proposed solution. Section 6 discusses the evaluations and results, while Section 7 concludes with insights and suggestions for future work.

2 BACKGROUND

Microservices architecture (Bai and Song, 2024) has become a cornerstone of modern cloud-native applications, enabling developers to break down monolithic systems into modular, independently deployable components. This paradigm offers several advantages, including greater agility in development, improved fault isolation, and the ability to scale individual components based on specific needs. However, these advantages come at the cost of increased complexity in managing the distributed nature of microservices. As each microservice often runs in its container and communicates with others over the network, issues such as inter-service latency, dependency management, and resource contention become critical challenges.

One major limitation in microservice-based architectures is the difficulty of efficiently allocating resources in dynamic and heterogeneous environments (Mishra et al., 2022). Traditional resource allocation methods often rely on static configurations or basic heuristics, which struggle to adapt to real-time workload variations and infrastructure changes. For example, latency-sensitive applications require resources to be allocated in proximity to minimize delays, while resource-intensive applications demand sufficient CPU and memory to maintain performance. Additionally, microservices often exhibit complex interdependencies, where the performance of one service impacts others, further complicating allocation decisions (Al-Debagy and Martinek, 2018).

Digital Twins (DTs) provide a transformative solution to address these challenges (Lombardo et al., 2022). By creating a virtual representation of physical resources, DTs enable continuous monitoring, simulation, and analysis of systems in real-time. In the context of data centers, DTs can model the behavior of servers, network links, and workloads, offering insights into system performance under varying conditions. This capability allows administrators to predict the impact of allocation decisions, optimize resource utilization, and proactively address potential bottlenecks or failures. Moreover, DTs can simulate "what-

if" scenarios to evaluate different allocation strategies before applying them to the physical infrastructure, reducing the risk of disruptions (Talasila et al., 2023).

Several techniques for resource allocation leveraging Digital Twins have been proposed. Rule-based approaches, for instance, use predefined policies to allocate resources based on metrics such as CPU usage or memory availability. While simple to implement, these methods often fail to capture the dynamic and interconnected nature of modern cloud-native environments (Raghunandan et al., 2023) (Alanezi and Mishra, 2023). Machine learning models, on the other hand, offer a more adaptive approach by learning patterns from historical data and making predictions about future resource demands. However, these models typically require significant computational resources and may struggle to incorporate real-time feedback.

Integrating Artificial Intelligence (AI) techniques, particularly Large Language Models (LLMs), into resource allocation frameworks has opened new avenues for innovation. LLMs, such as Llama3, excel at processing and reasoning over complex datasets, making them ideal candidates for decision-making in dynamic environments (Sharma et al., 2024). Combining the predictive capabilities of LLMs with the simulation power of Digital Twins makes it possible to create a highly adaptive and intelligent resource allocation system. This approach enables the consideration of multiple factors, including latency, bandwidth, and workload dependencies, to make optimal allocation decisions in real time. Furthermore, LLMs can incorporate contextual information, such as application priorities or user-defined constraints, to tailor decisions to specific requirements (Kim and Ben-Othman, 2023).

In this paper, we explore the integration of LLMs into a Digital Twin-based resource allocation framework for programmable data centers. By leveraging real-time metrics and advanced AI techniques, our proposed solution addresses the limitations of existing methods and demonstrates significant improvements in resource utilization, decision accuracy, and system performance.

3 PROBLEM DEFINITION

The problem of resource allocation for microservices in programmable datacenters can be formally defined as a constrained optimization problem. Let $S = \{s_1, s_2, \dots, s_n\}$ represent the set of servers in the datacenter, where each server s_i has a finite capacity for CPU, memory, and network bandwidth. Let

$W = \{w_1, w_2, \dots, w_m\}$ represent the set of microservice workloads to be allocated, where each workload w_j is defined by its resource requirements $r_j = (r_{j,\text{cpu}}, r_{j,\text{mem}}, r_{j,\text{bw}})$.

The goal is to allocate each workload w_j to a server s_i to maximize the overall system performance while respecting resource constraints. The objective function can be expressed as:

$$\text{Maximize } U(S, W) = \sum_{j=1}^m \sum_{i=1}^n x_{ij} \cdot u_{ij},$$

where x_{ij} is a binary variable indicating whether workload w_j is allocated to server s_i (1 if allocated, 0 otherwise), and u_{ij} is the utility gained from the allocation, which depends on factors such as latency, bandwidth, and server load.

Each server s_i must not exceed its available resources:

$$\sum_{j=1}^m x_{ij} \cdot r_{j,\text{cpu}} \leq C_{i,\text{cpu}},$$

$$\sum_{j=1}^m x_{ij} \cdot r_{j,\text{mem}} \leq C_{i,\text{mem}},$$

$$\sum_{j=1}^m x_{ij} \cdot r_{j,\text{bw}} \leq C_{i,\text{bw}},$$

where $C_{i,\text{cpu}}$, $C_{i,\text{mem}}$, and $C_{i,\text{bw}}$ represent the CPU, memory, and bandwidth capacities of server s_i .

Each workload w_j must be assigned to exactly one server:

$$\sum_{i=1}^n x_{ij} = 1, \quad \forall j \in \{1, 2, \dots, m\}.$$

Allocation decisions should minimize the inter-service latency for workloads with dependencies. Let d_{jk} represent the dependency between workloads w_j and w_k . The total latency L can be expressed as:

$$L = \sum_{j=1}^m \sum_{k=1}^m d_{jk} \cdot l_{ij,ik},$$

where $l_{ij,ik}$ is the latency between servers s_i and s_k hosting w_j and w_k , respectively.

The key decision variables in this optimization problem are:

- x_{ij} : Binary variables indicating the allocation of workloads to servers.
- u_{ij} : Utility values associated with each allocation.

This problem is NP-hard due to the combinatorial nature of the allocation and the dependencies between workloads. Traditional methods, such as heuristics or static rules, struggle to capture the problem's dynamic and interconnected nature. Furthermore, the real-time

Table 1: Summary of Related Work and Addressed Limitations.

| Approach | Focus Area | Limitations Addressed by Our Solution |
|--------------------------|--|--|
| (Van Huynh et al., 2020) | Digital twin approach for URLLC in edge networks | Adaptability to cloud-native microservice allocation |
| (Tchernykh et al., 2022) | Workload allocation for digital twins on clouds | Dynamic resource allocation for microservices |
| (Peng et al., 2022) | AI-driven offloading in smart industries | Generalization to cloud environments and microservices |
| (Li et al., 2024) | AI-driven resource allocation in cloud computing | Integration of digital twins with LLMs |

constraints imposed by cloud-native applications necessitate efficient and adaptive solutions.

To address these challenges, we leverage the predictive capabilities of Digital Twins and the reasoning power of Large Language Models (LLMs). Digital Twins provide a real-time, virtual representation of the data center, enabling continuous monitoring and simulation. LLMs, such as Llama3, process these rich datasets to make informed allocation decisions that balance resource utilization, minimize latency, and maximize system performance.

4 RELATED WORK

In recent years, integrating Digital Twins (DTs) and advanced resource allocation strategies has garnered significant attention in the context of cloud-native applications and programmable data centers. This section reviews pertinent literature, highlighting existing approaches and their limitations, which our proposed solution aims to address.

(Van Huynh et al., 2020) introduced a digital twin approach for ultra-reliable and low-latency communications (URLLC) in edge networks, focusing on optimal user association, task offloading, and resource allocation. While effective in edge scenarios, this method does not directly address the complexities of microservice allocation in cloud data centers.

Similarly, (Tchernykh et al., 2022) explored workload allocation for digital twins on clouds using low-cost microservices streaming interaction. Their work emphasizes cost efficiency but lacks consideration for dynamic resource allocation challenges inherent in cloud-native microservices.

(Peng et al., 2022) proposed distributed incentives for intelligent offloading and resource allocation in digital twin-driven innovative industries. Although their approach leverages AI for resource management, it is tailored to industrial applications and does not encompass the specific requirements of microservice architectures in cloud environments.

(Li et al., 2024) discussed efficient resource allocation in cloud computing environments using AI-driven predictive analytics. Their study provides valuable insights into AI-based resource management but does not explicitly focus on digital twin integration or microservice allocation challenges.

Table 1 provides a summary of the works presented. Existing solutions often exhibit limitations such as:

- **Lack of Adaptability:** Many existing approaches depend on static rules or predefined heuristics, making them unsuitable for the highly dynamic and unpredictable nature of cloud-native microservices. This lack of flexibility often results in suboptimal performance under varying workloads and rapidly changing infrastructure conditions.
- **Limited Scope:** A significant number of methods are designed for specific applications, such as industrial IoT or particular edge computing scenarios. While effective in narrow domains, these solutions struggle to generalize across diverse cloud-native environments with heterogeneous workloads, thereby limiting their broader applicability in programmable data centers.
- **Insufficient Integration:** Few solutions adequately integrate the real-time monitoring capabilities of digital twins with the advanced reasoning and predictive capabilities of Large Language Models (LLMs). This disconnect prevents the synergistic benefits that could arise from combining real-time infrastructure insights with intelligent resource allocation strategies.

To address these significant gaps, we propose a comprehensive and adaptive digital twin allocation framework that leverages the power of LLMs for programmable data centers. This approach introduces the following key innovations:

- **Dynamic Adaptability:** By harnessing the reasoning capabilities of LLMs, our framework enables real-time adjustments to workload allocations based on dynamic changes in infrastructure

states, workload demands, and inter-service dependencies. This ensures that resource management remains efficient and responsive under fluctuating conditions.

- **Comprehensive Integration:** Our solution seamlessly combines the physical modeling capabilities of digital twins with the intelligent decision-making of LLMs. This integration allows for holistic resource management, where both physical infrastructure metrics and workload requirements are considered in unison, resulting in more balanced and informed allocation decisions.
- **Enhanced Performance:** By optimizing resource utilization, minimizing inter-service communication latency, and balancing bandwidth demands, our framework significantly improves overall system performance. These improvements not only enhance application responsiveness but also contribute to reduced operational costs and increased sustainability in data center operations.
- **Scalability and Generalization:** Unlike existing methods, our approach is designed to operate effectively across a wide range of programmable data center environments. It generalizes well to heterogeneous workloads, making it suitable for multi-tenant cloud-native systems with varying application requirements.

5 DIRA-LDT

In order to address the challenges of dynamic resource allocation in programmable data centers for cloud-native microservices, we propose the Dynamic Intelligent Resource Allocation with Large Language Models and Digital Twins (DIRA-LDT) framework. DIRA-LDT integrates Digital Twins (DTs) with Large Language Models (LLMs) to optimize the allocation of microservices, considering real-time system states and workload demands.

The DIRA-LDT framework operates in three main phases: data collection and representation, decision-making via LLM, and real-time resource allocation. Below, we detail the core components and algorithms that govern the framework.

Digital Twins continuously monitors the physical data center infrastructure, collecting CPU utilization, memory usage, network latency, and bandwidth metrics. Each workload w_j is represented by its resource requirements $r_j = (r_{j,\text{cpu}}, r_{j,\text{mem}}, r_{j,\text{bw}})$ and any dependency relationships with other workloads. The LLM receives input on the current state of the data center and workload requirements. It processes these

inputs to generate an allocation decision x_{ij} for each workload w_j , indicating the optimal server s_i . The allocation decisions are executed, and Digital Twins updates the system state to reflect the new resource distribution.

The interaction between the Digital Twin system, the LLM, and the resource allocator is central to the functionality of the DIRA-LDT framework. Digital Twins are instantiated for each physical resource in the data center, such as servers, network links, and storage devices. These twins continuously collect real-time metrics and simulate the behavior of the physical systems. For each resource s_i , the Digital Twin generates a state vector \mathcal{S}_i containing information such as CPU utilization, memory usage, bandwidth availability, and latency metrics. These states are aggregated into a global state \mathcal{S} , the entire data center.

The global state \mathcal{S} and workload descriptions W are provided as input to the LLM. The LLM processes these inputs to evaluate candidate resource allocations. Using its reasoning capabilities, the LLM generates a set of recommended allocations $\{x_{ij}\}$, where each x_{ij} represents the assignment of workload w_j to server s_i .

The resource allocator module evaluates the recommendations provided by the LLM, ensuring that all constraints (e.g., resource capacities and inter-service latency) are satisfied. Allocations that meet the requirements are executed in the physical infrastructure, and the Digital Twins are updated to reflect the new system state.

The updated state \mathcal{S} is fed into the Digital Twin system, providing continuous monitoring for further optimization cycles.

5.1 Algorithm Description

To simplify the notation and enhance clarity, Table 2 summarizes the symbols and their definitions used in the algorithm.

Table 2: Symbol Definitions.

| Symbol | Definition |
|--------------------|--|
| \mathcal{S} | Set of servers in the datacenter |
| W | Set of workloads to be allocated |
| $r_{j,\text{cpu}}$ | CPU requirement of workload w_j |
| $r_{j,\text{mem}}$ | Memory requirement of workload w_j |
| $r_{j,\text{bw}}$ | Bandwidth requirement of workload w_j |
| $C_{i,\text{cpu}}$ | CPU capacity of server s_i |
| $C_{i,\text{mem}}$ | Memory capacity of server s_i |
| $C_{i,\text{bw}}$ | Bandwidth capacity of server s_i |
| x_{ij} | Binary variable indicating if w_j is assigned to s_i |
| u_{ij} | Utility of assigning w_j to s_i |
| l_{ij} | Latency between w_j and server s_i |
| d_{jk} | Dependency between workloads w_j and w_k |

The allocation process in DIRA-LDT is formalized in Algorithm 1. This algorithm begins by initializing all allocation variables x_{ij} to zero, indicating that no workloads have been assigned to any server. The Digital Twin system plays a crucial role in collecting real-time metrics from the data center, including CPU, memory, bandwidth usage, latency, and dependency information. These metrics are aggregated into a global state \mathcal{S} , representing the current state of the entire data center infrastructure.

Algorithm 1 DIRA-LDT Allocation Algorithm

Require: $\mathcal{S}, W, C_{i,\text{cpu}}, C_{i,\text{mem}}, C_{i,\text{bw}}, d_{jk}$

Ensure: Optimal allocation of workloads to servers.

```

1: Initialize  $x_{ij} \leftarrow 0$  for all  $i, j$ .
2: Obtain current system state  $\mathcal{S}$  from Digital Twins.
3: for each workload  $w_j \in W$  do
4:   Query LLM with  $\mathcal{S}$  and  $r_j$  to determine candidate
     servers  $s_i$ .
5:   for each candidate server  $s_i$  do
6:     if  $\sum_j x_{ij} \cdot r_{j,\text{cpu}} + r_{j,\text{cpu}} \leq C_{i,\text{cpu}}$ 
7:        $\sum_j x_{ij} \cdot r_{j,\text{mem}} + r_{j,\text{mem}} \leq C_{i,\text{mem}}$ 
8:        $\sum_j x_{ij} \cdot r_{j,\text{bw}} + r_{j,\text{bw}} \leq C_{i,\text{bw}}$  then
9:         Compute utility  $u_{ij} = \frac{1}{l_{ij} + \frac{1}{C_{i,\text{bw}} - r_{j,\text{bw}}}}$ .
10:        end if
11:      end for
12:      Assign  $w_j$  to  $s_i$  with maximum  $u_{ij}$ .
13:      Update  $\mathcal{S}$  via Digital Twins.
14:   end for
```

For each workload w_j in the set W , the framework queries the LLM with \mathcal{S} and the workload's resource requirements r_j . The LLM processes this input to generate a list of candidate servers s_i , ranking them based on their suitability for hosting the workload. Suitability is determined by evaluating resource availability and inter-service dependencies d_{jk} , ensuring that each server meets the resource constraints and can effectively minimize operational bottlenecks.

Once candidate servers are identified, the resource allocator computes a utility u_{ij} for each feasible allocation. This computation incorporates static metrics, such as server load and bandwidth availability, and dynamic metrics, such as inter-service latency. The workload w_j is assigned to the server s_i that maximizes u_{ij} , ensuring that the allocation aligns with the framework's goals of optimizing resource usage and minimizing delays.

After a successful allocation, the Digital Twin system updates the global state \mathcal{S} to reflect the changes in resource usage, including adjustments to CPU, memory, and bandwidth availability on the selected server. This feedback loop ensures that subsequent allocation decisions are informed by the most current state of the system.

By continuously integrating real-time data from

Digital Twins and leveraging the LLM's predictive capabilities, the DIRA-LDT framework achieves dynamic adaptability. This enables it to respond effectively to workload variability and infrastructure changes, ensuring optimal performance under diverse and dynamic conditions.

As shown in the evaluation section, the DIRA-LDT framework demonstrates significant improvements over existing methods by addressing the inherent challenges of dynamic resource allocation in programmable data centers. We can cite:

- **Dynamic Adaptability:** By leveraging the advanced reasoning and contextual understanding capabilities of Large Language Models (LLMs), the framework enables real-time adaptation to dynamic workloads and ever-changing system states. This adaptability ensures that resource allocation decisions remain optimal, even under unpredictable fluctuations in demand or infrastructure conditions. LLMs process real-time insights from Digital Twins, allowing the system to anticipate changes and proactively adjust allocations, resulting in improved efficiency and responsiveness.
- **Holistic Optimization:** The integration of Digital Twins within the framework allows for a unified view of both physical infrastructure and virtualized resources. This comprehensive approach enables precise decision-making by considering factors such as CPU utilization, memory consumption, bandwidth availability, and inter-service latency. By maintaining a real-time understanding of the entire data center, the framework achieves an optimized balance of resource utilization, minimizes bottlenecks, and ensures the effective execution of cloud-native applications.
- **Scalability:** DIRA-LDT is designed to operate seamlessly in large-scale data center environments, accommodating thousands of servers and workloads with diverse requirements. The framework's architecture is highly modular and scalable, allowing it to manage complex workloads while maintaining performance and stability. Whether handling small, latency-sensitive microservices or large-scale computational tasks, DIRA-LDT ensures consistent performance across varying scales of operation. Its ability to efficiently allocate resources in multi-tenant environments demonstrates its robustness and suitability for modern programmable data centers.

6 EVALUATION AND DISCUSSION

This section comprehensively evaluates the DIRA-LDT framework in a programmable data center environment, comparing it against four state-of-the-art techniques from the literature. The results demonstrate the superior performance of DIRA-LDT across multiple metrics. Insights and observations are derived from these evaluations to underline their advantages.

The evaluation scenario consists of a simulated data center with 50 servers equipped with 64 CPU cores, 256 GB of memory, and a 1 Gbps network link. The data center handles 100 microservice workloads with diverse resource requirements: CPU demands range from 1 to 16 cores, memory demands from 2 to 32 GB, and bandwidth requirements from 10 to 200 Mbps. Inter-service dependencies are defined for 30% of the workloads, emphasizing the need to minimize communication latency. The latency between servers varies from 1 ms for intra-rack communication to 20 ms for inter-rack communication, replicating realistic data center conditions. Additionally, the workloads are assigned varying deadlines, adding a layer of complexity to the evaluation as the allocation strategy must balance efficiency and responsiveness.

We evaluate the performance of DIRA-LDT using the following metrics: Allocation Accuracy (%), Average Latency (ms), Bandwidth Utilization (%), Workload Completion Rate (%), Decision Time (ms), and Resource Utilization Balance (RUB). These metrics are critical for understanding the effectiveness of resource allocation in terms of precision, efficiency, and responsiveness. DIRA-LDT is compared against the following baseline techniques:

- **Heuristic-Based Allocation (HBA):** This method uses a rule-based approach, relying on predefined static thresholds to make allocation decisions. These thresholds are typically derived from empirical data or domain knowledge, allowing the system to allocate resources based on simple rules. While this approach is computationally efficient, its static nature makes it unsuitable for handling dynamic and unpredictable workload patterns in modern data centers. As demonstrated in (Abdullah et al., 2017), heuristic-based methods often struggle to adapt to varying conditions, leading to suboptimal resource utilization and increased latency under fluctuating demands.
- **Statistical Model (SM):** This approach leverages predictive analytics to forecast resource demands and optimize allocation decisions. By using statistical techniques and historical data, the model pre-

dicts workload behavior and allocates resources accordingly. However, statistical models often lack the ability to capture complex patterns in highly dynamic environments, limiting their effectiveness. As highlighted in (Daradkeh and Agarwal, 2022), while these models provide an improvement over static heuristics, they still fall short in scenarios requiring real-time adaptability and deep contextual understanding.

- **Machine Learning (ML):** Supervised learning methods, trained on historical data, have been increasingly applied to resource allocation problems. These models learn patterns and correlations from past workloads and use this knowledge to predict future demands and allocate resources. While ML-based methods, such as those described in (Sun et al., 2021), offer greater flexibility and adaptability compared to statistical models, their reliance on historical data makes them susceptible to degradation in performance when encountering unforeseen workload patterns or infrastructure changes.
- **Deep Reinforcement Learning (DRL):** A dynamic and adaptive allocation strategy that uses reinforcement learning to optimize resource management. By interacting with the environment and learning from feedback, DRL models continuously improve their allocation decisions over time. As explored in (Shabka and Zervas, 2021), DRL offers significant advantages in handling dynamic and heterogeneous workloads. However, its computational complexity and training overhead pose challenges, particularly in scenarios requiring rapid decision-making or operating under strict latency constraints.

Figure 1 presents the allocation accuracy achieved by each technique. DIRA-LDT outperforms all baselines, achieving an accuracy of 98.5%, significantly higher than HBA (85.0%) and SM (88.3%). This improvement is attributed to integrating LLMs and Digital Twins, which enable DIRA-LDT to holistically consider resource constraints and inter-service dependencies, resulting in precise allocations. Higher accuracy directly translates to fewer rejected workloads, ensuring optimal use of the available infrastructure.

Regarding average latency, shown in Figure 2, DIRA-LDT achieves a mean latency of 5.3 ms, the lowest among all techniques. This is due to its capability to prioritize latency-sensitive workloads and allocate them to servers with minimal communication delays. Baseline techniques such as HBA and SM fail to adequately account for latency, resulting in higher values of 12.7 ms and 10.2 ms, respectively. The significant reduction in latency achieved by DIRA-LDT

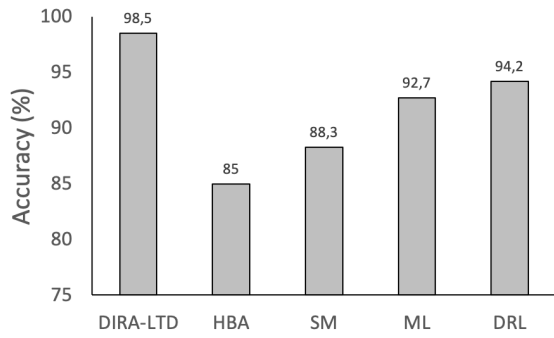


Figure 1: Comparison of Allocation Accuracy.

ensures that applications with strict performance requirements operate seamlessly.

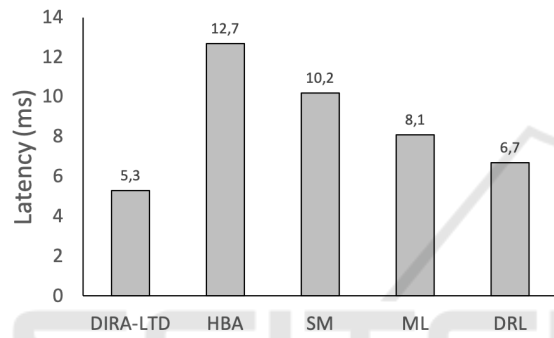


Figure 2: Comparison of Average Latency.

Bandwidth utilization results are presented in Figure 3. DIRA-LDT achieves 82.4% utilization, demonstrating its efficiency in balancing resource loads across the data center. In contrast, HBA and SM exhibit lower utilization rates of 70.5% and 75.8%, respectively, indicating suboptimal resource usage. The combination of LLM-driven decision-making and Digital Twins allows DIRA-LDT to maximize throughput without overloading servers. Efficient bandwidth utilization minimizes network contention, a critical factor in high-density data center environments.

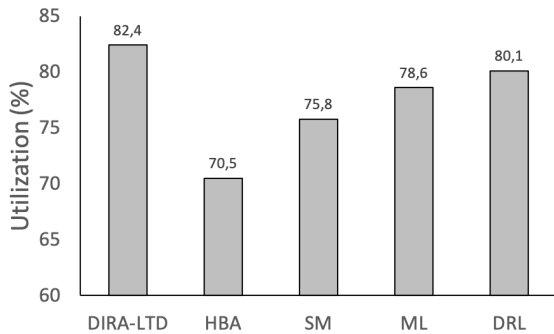


Figure 3: Comparison of Bandwidth Utilization.

Figure 4 illustrates the workload completion rate, a critical metric for latency-sensitive and real-time applications. DIRA-LDT achieves a completion rate of 96.3%, outperforming DRL (94.0%) and ML (91.2%). This highlights its effectiveness in meeting application deadlines through precise and adaptive allocation. Higher workload completion rates also reduce the likelihood of SLA violations, which can have financial and reputational implications for cloud providers.

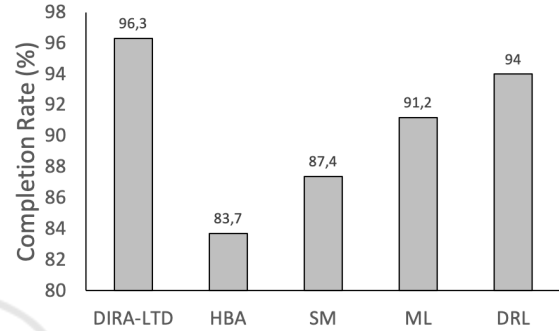


Figure 4: Comparison of Workload Completion Rate.

Decision time is an essential consideration for real-time systems. Figure 5 shows that DIRA-LDT requires 7.8 ms per allocation, slightly higher than HBA (3.2 ms) and SM (5.4 ms). However, significant improvements in other metrics justify the marginal increase in decision time, demonstrating the trade-off between computational overhead and allocation quality. Reduced operational inefficiencies offset the slight increase in decision time, making DIRA-LDT a viable option for real-world applications.

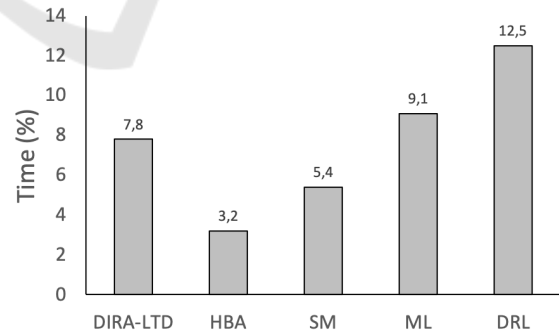


Figure 5: Comparison of Decision Time.

Finally, the Resource Utilization Balance (RUB) metric evaluates how evenly resources are distributed across the data center. A balanced allocation minimizes hotspots, reducing the likelihood of performance bottlenecks. DIRA-LDT achieves an RUB of 0.92 (on a scale of 0 to 1), significantly higher than HBA (0.78) and SM (0.84). This demonstrates its

ability to avoid resource overcommitment on a subset of servers, ensuring stability and predictability in multi-tenant environments.

The evaluation highlights several key insights. First, DIRA-LDT can adapt to dynamic workloads and infrastructure conditions, achieving superior allocation accuracy and workload completion rates. Second, its use of Digital Twins ensures a precise understanding of resource availability and interdependencies, leading to lower latency and higher bandwidth utilization. Third, while its decision time is marginally higher than that of more straightforward methods, the gains in accuracy and efficiency far outweigh the computational overhead. Lastly, the balance in resource utilization achieved by DIRA-LDT reduces the risk of server overloading, a common issue in traditional methods. Overall, DIRA-LDT represents a robust, scalable, and adaptive solution for resource allocation in programmable data centers, setting a new benchmark in performance and operational efficiency.

7 CONCLUSION

This paper has introduced the DIRA-LDT framework, a novel approach to resource allocation in programmable data centers that integrates Digital Twins and Large Language Models. The comprehensive evaluation of DIRA-LDT demonstrated its superiority over existing techniques in multiple performance metrics, including allocation accuracy, latency reduction, bandwidth utilization, workload completion rates, and resource utilization balance. By leveraging the unique capabilities of Digital Twins for real-time monitoring and LLMs for intelligent decision-making, DIRA-LDT has proven to be an adaptable, efficient, and scalable solution for managing the complexities of modern cloud-native environments.

The results highlight several significant contributions of the DIRA-LDT framework. First, it achieves a high allocation accuracy of 98.5%, far exceeding heuristic-based and statistical models. This ensures a more efficient utilization of resources while minimizing rejected workloads. Second, DIRA-LDT reduces average latency to 5.3 ms, a critical achievement for latency-sensitive applications. This improvement is enabled by the framework's ability to prioritize inter-service dependencies during allocation. Third, DIRA-LDT optimizes bandwidth utilization to 82.4%, maximizing throughput while avoiding overloading network links. Finally, it achieves a resource utilization balance of 0.92, ensuring that server loads are evenly distributed across the data center, thereby mitigating

risks of hotspots and enhancing overall stability.

Despite these promising results, the study also identified inevitable trade-offs. The decision time for DIRA-LDT is slightly higher than that for heuristic methods, reflecting the computational overhead of integrating LLMs into the decision-making process. However, this increase is marginal compared to the significant gains in accuracy, latency, and resource optimization. Future research could further optimize the decision time by employing lightweight LLM models or distributed processing techniques.

One area of future work is extending the framework's scalability. While the current study evaluated DIRA-LDT in a simulated environment with 50 servers and 100 workloads, real-world data centers often involve thousands of servers and highly dynamic workloads. Future experiments could assess the framework's performance in larger-scale scenarios, investigating how it adapts to increased complexity and workload variability.

Another promising avenue is integrating energy efficiency metrics into the allocation process. As data centers face growing pressure to reduce their carbon footprint, DIRA-LDT could be extended to consider energy consumption as a key objective. Digital Twins could simulate power usage at the server level, while LLMs could optimize allocations to balance performance and energy efficiency. This enhancement would make DIRA-LDT a performance-driven solution and an environmentally sustainable one.

Further research could also explore the integration of multi-cloud environments. In many scenarios, workloads are distributed across multiple data centers managed by different providers. Extending DIRA-LDT to operate in such heterogeneous environments would require incorporating inter-datacenter communication costs, latency, and security constraints into the decision-making process. This would position the framework as a versatile solution for managing resources in complex, distributed cloud ecosystems.

Additionally, the framework could benefit from more advanced learning capabilities. For instance, reinforcement learning could be combined with LLMs to enable the system to learn from historical allocation decisions and improve its strategies over time. This hybrid approach could enhance adaptability and optimize allocations in unpredictable or highly dynamic scenarios.

Finally, future work should explore real-world DIRA-LDT deployment in live data centers. While simulations provide valuable insights, real-world deployments would uncover practical challenges, such as integration with existing orchestration platforms like Kubernetes and OpenStack. Addressing these

challenges would validate the framework's practicality and provide opportunities to refine its design and functionality based on operational feedback.

In conclusion, DIRA-LDT represents a significant advancement in resource allocation for programmable data centers, combining the predictive power of LLMs with the real-time insights of Digital Twins. Its ability to outperform existing methods across multiple metrics underscores its potential to address the challenges of modern cloud-native environments. As data centers evolve, the adaptability and intelligence of DIRA-LDT provide a strong foundation for future innovation, paving the way for more efficient, sustainable, and intelligent resource management solutions.

ACKNOWLEDGEMENT

This work was supported by the Sao Paulo Research Foundation (FAPESP), grant 2023/00794-9. This study was partially funded by CAPES, Brazil - Finance Code 001.

REFERENCES

- Abdullah, M., Lu, K., Wieder, P., and Yahyapour, R. (2017). A heuristic-based approach for dynamic vms consolidation in cloud data centers. *Arabian Journal for Science and Engineering*, 42:3535–3549.
- Al-Debagy, O. and Martinek, P. (2018). A comparative review of microservices and monolithic architectures. In *2018 IEEE 18th International Symposium on Computational Intelligence and Informatics (CINTI)*, pages 000149–000154.
- Alanezi, K. and Mishra, S. (2023). Towards a scalable architecture for building digital twins at the edge. In *2023 IEEE/ACM Symposium on Edge Computing (SEC)*, pages 325–329.
- Bai, R. and Song, X. (2024). Research on information system architecture based on microservice architecture. In *2024 5th International Conference on Information Science, Parallel and Distributed Systems (ISPDS)*, pages 606–610.
- Buchaca, D., Berral, J. L., Wang, C., and Youssef, A. (2020). Proactive container auto-scaling for cloud native machine learning services. In *2020 IEEE 13th International Conference on Cloud Computing (CLOUD)*, pages 475–479.
- Daradkeh, T. and Agarwal, A. (2022). Cloud workload and data center analytical modeling and optimization using deep machine learning. *Network*, 2(4):643–669.
- Deng, S., Zhao, H., Huang, B., Zhang, C., Chen, F., Deng, Y., Yin, J., Dustdar, S., and Zomaya, A. Y. (2024). Cloud-native computing: A survey from the perspective of services. *Proceedings of the IEEE*, 112(1):12–46.
- Hongyu, Y. and Anming, W. (2023). Migrating from monolithic applications to cloud native applications. In *2023 8th International Conference on Computer and Communication Systems (ICCCS)*, pages 775–779.
- Kim, H. and Ben-Othman, J. (2023). Eco-friendly low resource security surveillance framework toward green ai digital twin. *IEEE Communications Letters*, 27(1):377–380.
- Li, H., Wang, S. X., Shang, F., Niu, K., and Song, R. (2024). Efficient resource allocation in cloud computing environments using ai-driven predictive analytics. *Applied and Computational Engineering*, 82:17–23.
- Li, H., Zhang, T., and Huang, Y. (2021). Digital twin technology for integrated energy system and its application. In *2021 IEEE 1st International Conference on Digital Twins and Parallel Intelligence (DTPI)*, pages 422–425.
- Lombardo, A., Morabito, G., Quattropani, S., and Ricci, C. (2022). Design, implementation, and testing of a microservices-based digital twins framework for network management and control. In *2022 IEEE 23rd International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, pages 590–595.
- Mishra, R., Jaiswal, N., Prakash, R., and Barwal, P. N. (2022). Transition from monolithic to microservices architecture: Need and proposed pipeline. In *2022 International Conference on Futuristic Technologies (INCOFT)*, pages 1–6.
- Mohammad, A. F., Clark, B., Agarwal, R., and Summers, S. (2023). Llm/gpt generative ai and artificial general intelligence (agi): The next frontier. In *2023 Congress in Computer Science, Computer Engineering, & Applied Computing (CSCE)*, pages 413–417.
- Morichetta, A., Pusztai, T., Vij, D., Pujol, V. C., Raith, P., Xiong, Y., Nastic, S., Dustdar, S., and Zhang, Z. (2023). Demystifying deep learning in predictive monitoring for cloud-native slos. In *2023 IEEE 16th International Conference on Cloud Computing (CLOUD)*, pages 1–11.
- Nguyen, N. T. and Kim, Y. (2022). *A Design of Resource Allocation Structure for Multi-Tenant Services in Kubernetes Cluster*.
- Peng, K., Huang, H., Bilal, M., and Xu, X. (2022). Distributed incentives for intelligent offloading and resource allocation in digital twin driven smart industry. *IEEE Transactions on Industrial Informatics*, 19(3):3133–3143.
- Raghunandan, A., Kalasapura, D., and Caesar, M. (2023). Digital twinning for microservice architectures. In *ICC 2023 - IEEE International Conference on Communications*, pages 3018–3023.
- Shabka, Z. and Zervas, G. (2021). Resource allocation in disaggregated data centre systems with reinforcement learning. *arXiv preprint arXiv:2106.02412*.
- Sharma, V., Sharma, K., and Kumar, A. (2024). Ai and digital twins transforming healthcare iot. In *2024 14th*

- International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, pages 6–11.
- Spyrou, A., Konstantinou, I., and Koziris, N. (2024). A decision support system for automated configuration of cloud native ml pipelines. In *2024 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops)*, pages 215–220.
- Sun, Y., Wang, X., Li, Z., and Zhang, Y. (2021). An efficient forecasting approach for resource utilization in cloud data centers using machine learning. *Neural Computing and Applications*, 33:12345–12358.
- Talasila, P., Gomes, C., Mikkelsen, P. H., Arboleda, S. G., Kamburjan, E., and Larsen, P. G. (2023). Digital twin as a service (dtaas): A platform for digital twin developers and users. In *2023 IEEE Smart World Congress (SWC)*, pages 1–8.
- Tchernykh, A., Facio-Medina, A., Pulido-Gaytan, B., Rivera-Rodriguez, R., and Babenko, M. (2022). Toward digital twins' workload allocation on clouds with low-cost microservices streaming interaction. *Computation*, 10(2):17.
- Van Huynh, D., Nguyen, V.-D., Khosravirad, S. R., Sharma, V., Dobre, O. A., Shin, H., and Duong, T. Q. (2020). Urllc edge networks with joint optimal user association, task offloading and resource allocation: A digital twin approach. *IEEE Journal on Selected Areas in Communications*, 38(8):1698–1718.

