INTERNATIONAL JOURNAL OF RESEARCH AND INNOVATION IN SOCIAL SCIENCE (IJRISS)
ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume IX Issue XII December 2025

Latency on Motion Synchronization in Game Engine-Driven Digital Twin Robotic Arms: Challenges and Techniques

Mohamad Lutfi Dolhalit., Nazreen Abdullasim*., Aiman Hakim Azahari., Mohd Khalid Mokhtar

Fakulti Teknologi Maklumat Komunikasi, Universiti Teknikal Malaysia Melaka, Melaka, Malaysia

DOI: https://dx.doi.org/10.47772/IJRISS.2025.91200037

Received: 11 December 2025; Accepted: 18 December 2025; Published: 31 December 2025

ABSTRACT

Ensuring precise and stable motion synchronization in game engine-driven digital twin robotic arms is challenging due to real-time data transmission delays. Various strategies—such as predictive modeling, network optimization, and robotic arm control techniques—have been proposed to address latency-aware motion synchronization. However, the absence of recent review papers in this area limits researchers' ability to identify the most effective solutions for minimizing latency and enhancing system reliability. To fill this gap, we conducted a systematic literature review (SLR) to identify, analyze, classify, and summarize existing latency reduction techniques. A total of 125 research studies from reputable sources were reviewed to uncover recent trends in the field. We developed a taxonomy to group the identified methods based on common characteristics and provided concise summaries of each approach. Furthermore, this study outlines key research challenges and suggests future directions for improving synchronization accuracy. The findings offer a comprehensive and structured overview of existing solutions and serve as a valuable reference for researchers and practitioners aiming to advance real-time digital twin applications through game engine-based visualization.

Keywords: Latency, Motion Synchronization, Digital Twin, Game Engine, Data Transmission, Robotic

INTRODUCTION

Digital Twin (DT) technology has become a key enabler in various industries by providing real-time monitoring, prediction, and control through virtual replicas of physical systems (Parashar & Gameti, 2024; Tao et al., 2019). Game engines, such as Unity and Unreal Engine, are increasingly adopted for digital twin development due to their advanced simulation and visualization capabilities, especially in robotic applications (Lin et al., 2022; Haque et al., 2023).

However, a major challenge in this domain is ensuring accurate and real-time synchronization between the virtual model and its physical counterpart, particularly in robotic systems where latency can lead to motion discrepancies and reduced reliability in tasks like teleoperation and human-robot collaboration (Zheng et al., 2021; Zhang et al., 2022). In recent years, researchers have proposed various solutions to mitigate latency, including predictive algorithms (Wang et al., 2022), interpolation techniques (Ali et al., 2023), and network-level optimizations, such as edge computing and communication protocol enhancements (Cheng et al., 2023). While these methods have improved system responsiveness, they still face limitations, such as reduced accuracy under variable conditions, dependency on network infrastructure, and a lack of adaptability to complex real-time interactions. Moreover, there is a scarcity of comprehensive reviews that critically analyse and compare these techniques within the specific context of game engine-driven digital twin robotics (Lee et al., 2023; Hussain et al., 2022). To address this gap, this study conducts a systematic literature review (SLR) with three main contributions: (1) identifying and classifying existing latency mitigation techniques, (2) evaluating their effectiveness in real-time robotic applications involving game engines, and (3) proposing a taxonomy that organizes these methods based on shared characteristics. Rather than introducing a new technique, this work enhances current understanding by synthesizing and summarizing recent studies, offering a structured reference for researchers and developers. The rest of the paper is organized as follows: Section 2 details the SLR methodology, Section 3 presents the taxonomy and classification of latency mitigation methods, Section 4 discusses findings and ongoing challenges, and Section 5 concludes with future research directions.

Related Work

Method

To ensure a high-quality and substantive literature review, a systematic literature search was conducted, dividing the identification of relevant literature into three key steps: Literature Search, Literature Selection, and Qualitative Analysis.

LITERATURE SEARCH

Following the systematic literature review (SLR) method proposed by Webster and Watson (2002), we began by identifying relevant studies on digital twin technology, with a specific focus on latency-aware synchronization in game engine-driven robotic applications. The initial search was conducted on Google Scholar using broad keywords, such as "digital twin", "game engine", "robotics", and "latency", combined with Boolean operators (e.g., "digital twin" AND "latency", "game engine" AND "robotic arm", "digital twin" AND "predictive synchronization"). We then refined the search using more focused terms including "motion synchronization", "interpolation", "latency compensation", "predictive algorithm", "edge computing", and "real-time control", to capture specific techniques and challenges addressed in recent studies. Academic databases, such as IEEE Xplore, SpringerLink, ScienceDirect, and EBSCOhost were queried, and only peer-reviewed journal articles or conference proceedings published in English between 2018 and 2024 were considered. Inclusion criteria required the presence of relevant keywords in the title, abstract, or keywords section. After screening titles and abstracts, we shortlisted 125 papers and documented metadata, such as DOI, citation count, publication venue, and research domain. Additionally, the "Connected Papers" tool was used to trace influential and closely related studies based on co-citation networks, ensuring the coverage of both foundational and recent contributions in the field.

Literature Selection

To ensure a manageable and high-quality selection of studies for this review, a systematic filtering process was implemented. Initially, duplicate records retrieved from multiple databases were removed. Titles and abstracts were then screened to exclude publications that were unrelated to digital twin technologies or lacked relevance to latency-aware motion synchronization in robotic systems. However, papers featuring notable application examples were selectively retained. To ensure academic rigor, the quality of publications was assessed using the Verband der Hochschullehrer für Betriebswirtschaft (VHB) JOURQUAL ranking system, which is widely adopted in the information systems and business research communities. Preference was given to journal articles ranked B and C, while exceptions were made for highly cited works that demonstrated significant relevance regardless of their ranking. As a result, fifteen core papers were selected for in-depth analysis. These papers were then systematically classified into three thematic streams implementation frameworks, synchronization techniques, and predictive algorithms based on their primary contributions to latency mitigation in digital twin robotic systems. This categorization is presented in Table 1, which maps each study’s methodological focus, application domain, and observed trade-offs. The structured classification highlights both well-established solutions and underexplored areas, such as the absence of integrated hybrid AI-edge synchronization frameworks.

Qualitative Analysis

Before presenting the results, we first outline the approach used to systematically analyse the selected literature. To ensure a comprehensive and structured evaluation, we adopted the qualitative analysis method proposed by Wolfswinkel et al. (2013), which consists of three stages: Open Coding, Axial Coding, and Selective Coding. In the Open Coding phase, we extracted key concepts, methods, and application themes related to latency mitigation in digital twin robotic systems. During the Axial Coding stage, we grouped similar studies based on methodological focus and their contribution to latency reduction. Finally, in the

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume IX Issue XII December 2025

Selective Coding phase, these groupings were refined into three core thematic streams: implementation frameworks, synchronization techniques, and predictive algorithms. These categories form the basis for the analytical classification presented in Table 1, which maps the focus, techniques, and trade-offs across the selected studies to reveal current trends and research gaps in latency-aware digital twin applications.

Table 1. Analytical classification of latency mitigation studies in digital twin robotic systems across implementation frameworks, synchronization techniques, and predictive algorithms

Stream	**Study**	**Focus**	**Method**	**Trade-offs / Gaps**
Implementation Frameworks	Garrido-Hidalgo et al., 2022	Predictive Maintenance via DTs	Data-driven models	Limited scalability in heterogeneous systems
	Mrzyk et al., 2023	Modular DT architecture	Flexible IT-infrastructure	No real-time feedback loop
Synchronization Techniques	Tan et al., 2023	DT sync problem formulation	Sync protocols + framework	Lack of performance benchmarks
	Shen et al., 2025	Real-time update for DTs	Update algorithms	Not evaluated under high-load scenarios
Predictive Algorithms	Liu et al., 2022	ML-based Predictive Maintenance	Convolutional Autoencoders	High computational demand
	Polese et al., 2018	Latency prediction in 5G networks	ML + Traffic Routing	Needs real-time adaptability

METHODOLOGY

Digital Twin Technology Overview

Digital Twin (DT) technology enables real-time monitoring, analysis, and optimization by creating digital replicas of physical systems (Tao et al., 2019; Fuller et al., 2020). A digital twin is a dynamic model that continuously interacts with its physical counterpart through three key components: the digital model, the physical entity, and the data connection (Boschert & Rosen, 2016). Efficient data transmission and synchronization within this system are essential for informed decision-making and operational accuracy (Yin et al., 2022; Kaur & Mishra, 2022). The concept of the digital twin was originally introduced by Michael Grieves in 2002 within the Product Lifecycle Management (PLM) paradigm (Grieves & Vickers, 2017) and has since evolved into a transformative tool across multiple domains. Digital twins have been widely adopted in industries, such as urban planning, manufacturing, healthcare, and aerospace, enabling applications like real-time asset monitoring, predictive maintenance, and virtual testing (Rehman et al., 2022; Liu et al., 2020; Xu et al., 2021; Jones et al., 2020). In robotics, digital twins facilitate state estimation, behavior prediction, and lifecycle analysis, contributing to increased system autonomy and reduced downtime (Zhang et al., 2021; Lee et al., 2014). This capability supports complex simulations and scenario testing without disrupting real-world operations, making digital twins increasingly valuable in the development of smart, interconnected systems (Alam, 2023). Table 2 provides an overview of Digital Twin Technology and its key components.

Table 2. Overview of digital twin technology and its key components

Aspect	**Details**
Definition	A digital twin is a dynamic and continuously evolving virtual model that precisely mirrors a physical entity, enabling real-time monitoring, simulation, and optimization of its performance across its entire lifecycle (Modoni et al., 2023; Ünal et al., 2023).

	It consists of three main components: the physical entity, the virtual model, and the data connections that allow for seamless interaction between them (Boschert & Rosen, 2016; Fuller et al., 2020). By utilizing real-time data, digital twins not only reflect the current state of their physical counterparts but also predict their future behaviour, ultimately improving decision-making and operational efficiency (Tao et al., 2019; Lee et al., 2014; Grieves & Vickers, 2017). This capability has led to their increasing adoption across sectors, such as manufacturing, aerospace, smart cities, and healthcare, due to their ability to enhance operational insight, enable predictive maintenance, and support complex simulations without disrupting real-world operations (Liu et al., 2020; Jones et al., 2020).
Origins	The concept of digital twins was first introduced by Michael Grieves in 2002 during a presentation at the University of Michigan, where he discussed a virtual model that mirrors a physical counterpart within Product Lifecycle Management (PLM) (Grieves & Vickers, 2017). Earlier applications of similar concepts can be traced back to NASA during the Apollo missions in the 1960s, where simulators were used to model spacecraft conditions.
Applications	Smart manufacturing Digital twin technology enhances real-time monitoring, predictive maintenance, and system optimization in smart manufacturing, significantly reducing machine downtime, energy consumption, and production errors (Tao et al., 2022; Liu & Zhang, 2021; Qi & Tao, 2018). In the aerospace industry, digital twins are used for performance monitoring, structural health diagnostics, and mission simulation, thereby improving operational safety and reducing maintenance costs (Fuller et al., 2020; Glaessgen & Stargel, 2012). Key applications include product lifecycle management, dynamic design optimization, and intelligent control systems (Grieves & Vickers, 2017; Boschert & Rosen, 2016). Despite these advantages, several challenges persist. These include high implementation costs, data interoperability issues across platforms, cybersecurity risks due to continuous data exchange, and the need for standardized frameworks (Khan et al., 2021; Ünal et al., 2023). Conversely, ongoing research presents opportunities, such as integrating AI for autonomous decision-making, utilizing 5G/6G for ultra-low latency communication, and expanding the use of digital twins in emerging domains like personalized medicine and urban digital infrastructure (Lu et al., 2020; Jones et al., 2020; Alam, 2023). Healthcare Digital twins in healthcare facilitate personalized treatments through patient-specific simulations, significantly enhancing diagnostic accuracy, treatment planning, and clinical decision-making—thereby advancing the field of precision medicine (Kahn & Lentz, 2024; Rojas & Gutiérrez, 2024; Bruynseels et al., 2018). Key applications include surgical simulations, which allow for preoperative rehearsals and risk assessment; treatment planning, where simulations model disease progression and therapy responses; and real-time patient monitoring, enabling continuous assessment through sensor-integrated twins (Corral-Acero et al., 2020; Björnsson et al., 2020; Li et al., 2021).Despite their potential, digital twins in healthcare face several challenges, such as ensuring data privacy and security, integration of heterogeneous medical data, limited standardization in clinical environments, and the high cost of implementation (Marr, 2022; Fernandes et al., 2021; Yang et al., 2022). Moreover, achieving clinically validated, real-time predictive models remain an open research problem due to the complexity and variability of human physiology. However, ongoing advancements in AI, wearable technology, and high-performance computing present

	promising opportunities to overcome these hurdles and enable real-time, adaptive, and patient-centred healthcare systems (Tao et al., 2022; Alam, 2023). Urban Planning and Smart Cities Digital twins play a crucial role in both urban planning and smart city applications by leveraging real-time data for improved decision-making and resource allocation. In urban planning, they are used for simulating zoning changes, visualizing infrastructure development, and optimizing land use strategies (Ciorra & De Rosa, 2022; Karam & Alshahrani, 2022). In smart city applications, digital twins help optimize traffic management, energy distribution, public safety systems, and infrastructure maintenance by integrating data from IoT devices and city-wide networks (Kahn & Lentz, 2024; Lin et al., 2023; Batty, 2018). These technologies enhance real-time monitoring and predictive analytics to improve service delivery, although challenges remain in ensuring equitable data governance, interoperability, and citizen privacy (Karam & Alshahrani, 2022; Lin et al., 2023).
Key components	Digital twins provide a real-time reflection of physical entities by offering a highly synchronized representation of their status and behaviours (Modoni et al., 2022; Fuller & Barlow, 2019). Their dynamic interaction enables continuous, bidirectional communication between the physical object and its digital counterpart, ensuring accurate modelling and simulation (Liu & Zhang, 2023). Through self-evolution, digital twins adapt and optimize over time based on real-time data from their physical counterparts, allowing for continuous improvement (Modoni et al., 2022). Additionally, each physical entity must have a unique digital twin that evolves throughout its lifecycle, ensuring identifiability (Fuller & Barlow, 2019). Digital twins also enhance predictive capabilities by forecasting the behaviour and performance of physical counterparts, aiding in decision-making processes (Liu & Zhang, 2023). Lastly, they rely on data integration from multiple sources to create a comprehensive model that supports analysis and simulation (Modoni et al., 2022; Liu & Zhang, 2023).
Benefits	Digital twin technology enhances predictive maintenance by analysing real-time sensor data to detect issues before failures occur, allowing organizations to schedule maintenance proactively, reduce downtime, and extend equipment lifespan (Yang et al., 2017). It also improves operational efficiency by optimizing robotic performance and industrial processes, enabling continuous monitoring and real-time adjustments that lead to increased productivity (Fuller & Barlow, 2019). Additionally, digital twins support simulation and testing, allowing organizations to conduct experiments and validate designs in a virtual environment without real-world risks. This capability helps optimize processes before implementation, reducing costs and improving overall system performance (Modoni et al., 2022).

Digital Twin for Robotic Arms

Digital twin (DT) technology is pivotal in robotics, especially for controlling and manipulating robotic arms. By creating a virtual replica of a robotic system, DTs enable real-time monitoring and simulation of movements, ensuring precision and safety during operations. Utilizing data from multiple embedded sensors, DTs support advanced collision detection and movement analysis, enhancing operational efficiency and enabling predictive maintenance (Zong et al., 2021; Du et al., 2021). The applications of DTs in robotics are expanding rapidly. Industries increasingly employ them for offline programming, allowing engineers to simulate and test various scenarios in a controlled virtual environment before real-world implementation. This approach reduces risks related to physical trials, improves accuracy, and enhances safety (Gallala et al., 2022). Recent research explores the integration of artificial intelligence (AI) with DT frameworks to improve robotic

INTERNATIONAL JOURNAL OF RESEARCH AND INNOVATION IN SOCIAL SCIENCE (IJRISS)
ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume IX Issue XII December 2025

arm control strategies. For instance, Zhang et al. (2025) developed a high-fidelity simulation platform for robot dynamics in a DT environment, facilitating accurate control logic and simultaneous testing of real and simulated environments. Li and Yang (2025) discuss how integrating DTs with embodied AI can bridge the sim-to-real gap, transforming virtual environments into dynamic platforms for training and optimization. As more industries adopt this technology, its potential to drive intelligent automation and optimize production processes continues to grow. However, challenges remain, including ensuring real-time synchronization between virtual and physical systems, managing the complexity of high-fidelity models and addressing potential latency in data transmission (Liang et al., 2022; Liu et al., 2023).

Integration of Game Engines in Digital Twin Systems

The integration of game engines into digital twin systems has gained significant attention due to their ability to create interactive, immersive, and real-time visual simulations (Wang et al., 2024; Zhou et al., 2023; Zhang et al., 2022). These platforms enable real-time monitoring of physical twins within 3D environments, including virtual reality (VR) and augmented reality (AR), allowing users to engage with complex simulations. This capability is particularly valuable in industrial applications that require precise visualization and interaction, such as robotics and aerospace engineering (Liu et al., 2023). One of the key advantages of using game engines in digital twin robotics is their ability to enhance metric visualization, improve simulation accuracy, and provide near-photorealistic rendering (Rundel & De Amicis, 2023; Kim et al., 2022). These features allow engineers to conduct detailed performance evaluations and optimize robotic workflows before real-world deployment. Additionally, game engines facilitate the simulation of large datasets, including environmental and geographical parameters, which are essential for applications in urban planning, autonomous systems, and industrial automation (Chen et al., 2024; Alshammari et al., 2023). Despite these benefits, game engine-driven simulations come with significant computational demands. High-fidelity simulations require substantial processing power, which can limit real-time applications in resource-constrained environments. To address these challenges, future research should explore optimization techniques such as adaptive level-of-detail rendering, GPU offloading, and cloud-based computation (Huang et al., 2023; Xu et al., 2022). Figure 1 illustrates an integrated digital twin framework for real-time visualization and optimization of robotic arm operations using game engine technology.

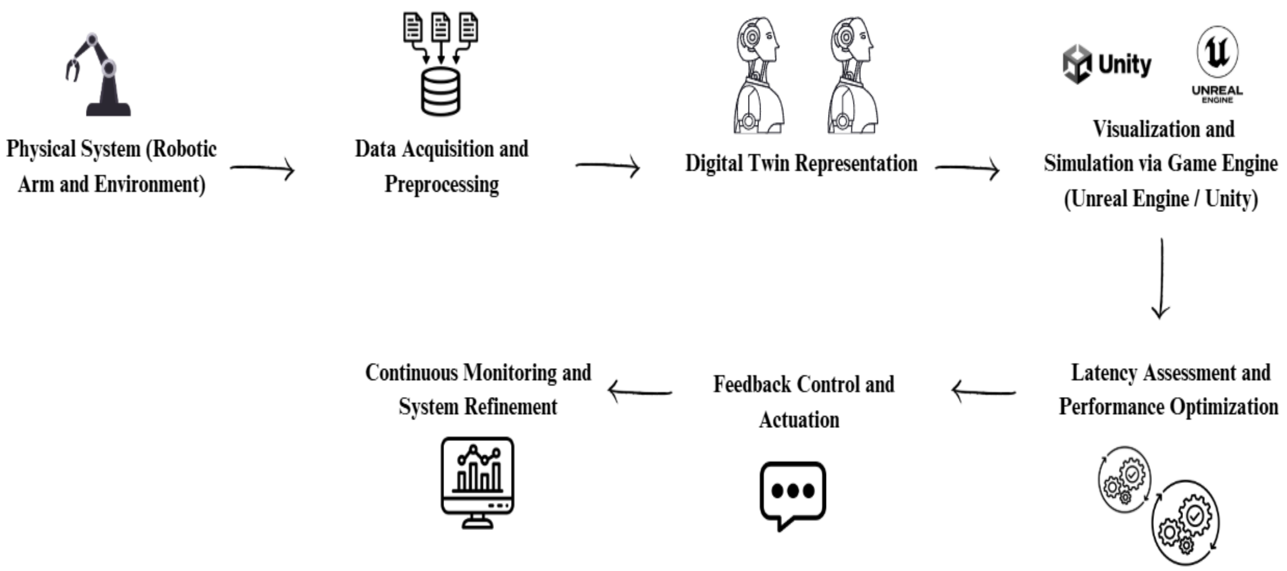

Figure 1. An integrated digital twin framework for real-time visualization and optimization of robotic arm operations using game engine technology.

Motion Synchronization Between Game Engine and Robotic Arm

Achieving precise motion synchronization between game engines and robotic arms is essential for real-time digital twin applications. Game engines, such as Unity and Unreal Engine, enable seamless integration by processing data from robotic arm encoders, tracking position and orientation, and simulating the arm's

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume IX Issue XII December 2025

movements within a 3D environment (Rundel & De Amicis, 2023; Zhang et al., 2023). This real-time synchronization is crucial in applications such as industrial automation, remote robotic control, and virtual training environments (Chen et al., 2024). To enhance motion synchronization, recent research has explored the use of motion capture systems integrated with Unreal Engine to allow operators to control a handheld device that mimics the robotic arm's end-effector. As the operator moves the device, the virtual representation of the robotic arm updates in real-time, providing immediate visual feedback and enabling more intuitive control (Kumar et al., 2024; Liet al., 2023). This approach significantly improves the precision and usability of robotic systems, particularly in teleoperation and augmented reality-based robotic training.

One of the primary challenges in motion synchronization is latency, which affects real-time interactions and system responsiveness. Predictive motion modelling techniques, such as linear extrapolation and Kalman filtering, have been applied within Unity to estimate future positions of the robotic arm based on current movement trends, effectively reducing perceived delay (Wang et al., 2023). Meanwhile, latency compensation algorithms, such as time-delay estimation and dynamic adjustment of rendering frames, have been used in Unreal Engine to align visual feedback with delayed physical responses (Tan et al., 2024). These methods aim to reduce motion discrepancies and improve synchronization accuracy. However, further experimental validation is needed to determine the robustness and adaptability of these techniques across different operational environments and hardware configurations.

Challenges in Latency

Despite significant advancements in latency reduction, achieving precise motion synchronization in robotic systems continues to pose critical challenges. Unstable network conditions often introduce unpredictable delays that compromise the reliability of real-time synchronization (Lee et al., 2024). Moreover, integrating hardware components from different manufacturers can result in system-level inconsistencies, affecting performance and interoperability (Siciliano & Khatib, 2016). These challenges are particularly evident in collaborative robotics, where precise coordination is vital for tasks such as dual-arm manipulation or cooperative handling of heavy or delicate objects. (Moysis et al.,2020) Previous work also emphasizes the complexities involved in maintaining synchronization in such scenarios, underlining the need for robust latency mitigation strategies. Table 3 summarizes the primary challenges impacting latency and performance in robotic systems, highlighting the technical barriers that must be addressed to ensure seamless and safe real-time operation.

Table 3. Challenges affecting latency and performance in robotic systems.

Challenge	**Description**	**Impact on Performance**
Variability in Network Conditions (Lee et al., 2024)	Unpredictable delays caused by fluctuating network conditions	Complicates synchronization efforts
Integration of Diverse Hardware (Moysis et al., 2020)	Inconsistencies arising from using hardware components from different manufacturers	Affects overall system performance
Coordination of Multiple Robots (Moysis et al., 2020)	Requires precise synchronization when multiple robots handle large objects	Increases complexity and potential for errors

Latency Constraints in Different Synchronization Approaches

Different synchronization techniques introduce varying latency constraints, affecting the efficiency of robotic systems. Passive synchronization algorithms, for example, aim to align sensor data with robotic controllers but often introduce timing inconsistencies due to network transmission delays. A notable challenge is seen in unmanned aerial vehicles (UAVs), where low-latency synchronization is essential for guidance, navigation, and control (GNC) systems. Recent studies highlight the use of exponential moving average filters to mitigate latency offsets and synchronize clock drift between flight controllers and companion computers, enabling more stable communication and sensor data alignment (Gonzalez et al., 2023). In multi-robot systems, latency constraints can disrupt cooperative tasks, such as trajectory tracking and obstacle avoidance, emphasizing the need for efficient time-sensitive synchronization techniques (Bañez et al., 2017). Addressing these constraints through predictive modelling and real-time feedback mechanisms can significantly improve robotic coordination and motion accuracy.

Techniques for Latency Minimization in Motion Synchronization

Reducing latency in motion synchronization is crucial for enhancing the performance of robotic systems. One effective approach is network optimization, which minimizes data transmission delays to enable near real-time interaction. Advanced techniques, such as dynamic routing algorithms and adaptive packet prioritization, help reduce the time required for control commands to be transmitted and processed (Li et al., 2021; Ahmed & Kim, 2022). Dynamic routing enables routers to adjust communication paths based on current network conditions, ensuring efficient data flow (Li et al., 2021). Adaptive packet prioritization ensures that critical control commands are transmitted with higher priority, reducing latency in time-sensitive applications (Ahmed & Kim, 2022). Additionally, edge computing has emerged as a key latency minimization strategy, allowing data processing to occur closer to the robotic system and significantly decreasing round-trip latency (Wang et al., 2023). By processing data at the network's edge, systems can respond more quickly to real-time events, which is essential for applications like autonomous vehicles and industrial automation (Zhou et al., 2021).

Combining these network optimization methods with predictive algorithms and machine learning (ML) based latency compensation can further improve real-time performance in robotic applications. For instance, latency-aware collaborative perception systems use ML models to synchronize asynchronous data streams, enhancing the robustness and effectiveness of multi-agent robotic systems (Chen et al., 2022).

Network Optimization for Lower Latency

Network optimization plays a crucial role in minimizing latency in robotic systems, particularly in environments where real-time communication is essential. Research has shown that enhancing network pathways can significantly reduce communication delays, allowing robotic systems to operate more efficiently in dynamic settings (Lee et al., 2024). Advanced techniques, such as Quality of Service (QoS) management and adaptive bandwidth allocation, are commonly employed to improve responsiveness, ensuring that critical data packets receive priority during transmission (Lee et al., 2024). These strategies help maintain reliable and low-latency communication, which is essential for real-time robotic operations. Table 4 presents key techniques for network optimization aimed at reducing latency and enhancing system performance.

Table 4. Techniques to reduce latency in terms of network optimization for lower latency.

Section	**Techniques**	**Description**
Network Optimization for Lower Latency	Multi-Network Latency Prediction (Balota et al., 2023)	Utilizes linear interpolation and extrapolation algorithms to predict end-to-end latency in IoT networks, enhancing synchronization for robotic movements.
	Unity and ROS Integration (Singh et al., 2024)	Combines Unity with Robot Operating System (ROS) to enhance communication layers, achieving approximately 77.67 ms latency between commands and actions in robotic arms.
	Quality of Service Optimization (Phadke, J.,2023).	Prioritizes critical network traffic to minimize latency for latency-sensitive applications.

Predictive Algorithms and Interpolation Methods

Predictive algorithms and interpolation techniques are essential for mitigating latency issues in motion synchronization by leveraging past data to anticipate future conditions. This approach allows robotic systems to compensate for delays when executing commands, ensuring smoother and more accurate movements. In

2009, Howard introduced a model-predictive trajectory generation approach that improves path tracking for mobile robots, enabling proactive rather than reactive adjustments. Additionally, interpolation methods such as spline and linear interpolation help generate fluid movement trajectories by considering potential delays, thereby enhancing overall accuracy and synchronization (Howard, 2009). These techniques play a crucial role in optimizing robotic performance, as summarized in Table 5, which outlines key methods for reducing latency through predictive algorithms and interpolation.

Table 5. Techniques to reduce latency in terms of predictive algorithms and interpolation methods.

Section	**Techniques**	**Description**
Predictive Algorithms and Interpolation Methods	Interpolation Techniques (Xu et al., 2021)	Employs interpolation methods to estimate intermediate values based on known data points, smoothing transitions and reducing perceived latency in robotic motion synchronization.
	Soft Actor-Critic (SAC) Reinforcement Learning (Zhang et al., 2022)	Integrates SAC with digital twin technology for adaptive learning, allowing robotic arms to improve motion predictions based on previous interactions, effectively reducing latency through continuous feedback loops.
	AI based ML Program (Polese et al., 2018)	Employs ML to predict network latency and optimize traffic routing in 5G cloud computing, improving network performance and reducing latency.

Hardware and Software Co-Optimization Strategies

Co-optimization strategies for hardware and software are essential for reducing latency in robotic systems, ensuring both components work together effectively to enhance performance. This approach involves adjusting hardware configurations, such as sensor positioning and actuator sensitivity, while simultaneously optimizing software algorithms for faster processing efficiency (Lee et al., 2024). Research has shown that combining high-performance computing resources with efficient software frameworks can significantly reduce latency, improving real-time responsiveness in robotic applications (Lee et al., 2024). Table 6.0 presents key techniques for minimizing latency through hardware and software co-optimization strategies.

Table 6. Techniques to reduce latency in terms of hardware and software co-optimization strategies.

Section	Techniques	Description
Hardware and Software Co-Optimization Strategies	Adaptive Rendering Techniques (Wang et al., 2022)	Implements adaptive quality settings that dynamically change rendering resolution based on performance metrics, helping maintain lower latency during robotic operations.
	Feedback Mechanisms (Wang et al., 2022)	Utilizes robust feedback systems analyzing both subjective user experiences and objective performance metrics (like frame rates and latency) to enhance hardware-software interactions, leading to improved synchronization.
	UNICO Framework for AI Accelerators (Rashidi, B., Gao, C., Lu, S., Zhisheng, W., Wei, L., & Jui, S. ,2023).	Employs multi-objective Bayesian optimization to co-optimize hardware architectures and software mapping for deep neural networks, enhancing robustness and generalizability.

Comparative Analysis of Techniques

A comparative analysis of latency reduction methods reveals that no single approach is universally optimal; rather, the effectiveness of each technique depends on the specific application context and operational

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume IX Issue XII December 2025

requirements. For instance, while network optimization offers significant benefits in teleoperated environments where minimizing communication delays is critical, predictive algorithms may be more advantageous in scenarios demanding rapid response times (Zhan et al., 2023). Furthermore, integrating multiple strategies like combining network optimization with predictive algorithm can create synergistic effects that enhance overall system performance. Table 7 presents a comparison of various latency reduction techniques in robotic systems, highlighting their strengths and applicability in different use cases.

Table 7. Comparison of latency reduction techniques in robotic systems.

Technique	**Application context**	**Advantage**	**Disadvantage**
Network Optimization (Cundar et al., 2023)	Teleoperation	Reduces communication delays	May require complex infrastructure
Predictive Algorithms (Kumari et al., 2023)	Rapid response environments	Enhances responsiveness	May introduce inaccuracies if predictions fail
Combined Strategies (Scheer et al., 2023)	Various scenarios	Synergistic effects on performance	Complexity in implementation
Hardware and Software Co-Optimization (Aygün et al., 2023) (Zhang & Wang, 2021) (Dniu et al., 2023)	High-performance computing	Improves resource utilization and efficiency	Requires tight integration between hardware and software

Future Direction

Future research should explore the development of a Latency Compensation Framework (LCF) that integrates predictive modelling, real-time feedback loops, and adaptive network optimization to further minimize latency in digital twin robotic systems (Yang et al., 2024; Scheer et al., 2023). By dynamically adjusting synchronization parameters based on real-time performance metrics, the LCF can enhance responsiveness and system robustness, ensuring seamless transitions between virtual and physical models in complex robotic environments (Yang et al., 2024). Moreover, incorporating AI-driven self-learning compensation mechanisms into the LCF could further optimize latency mitigation strategies over time, improving motion synchronization accuracy and system efficiency (Kumari et al., 2023). These advancements will be essential in achieving high-precision digital twin applications for robotics, automation, and teleoperation.

To address these challenges, future research should also focus on developing more accurate and reliable latency measurement techniques. One promising direction is to explore microcontroller-based solutions to enhance precision in latent detection and monitoring. By leveraging the flexibility and real-time processing capabilities of microcontroller platforms, researchers can design customized latency measurement systems tailored to the specific requirements of digital twin applications (Kumar et al., 2022).

Additionally, further investigation into adaptive predictive algorithms, particularly reinforcement learning approaches, could optimize motion synchronization by dynamically adjusting parameters in response to changing network conditions, thereby minimizing the impact of latency and enhancing overall system performance (Chen et al., 2023).

Moreover, the integration of AI and ML for dynamic latency compensation presents a promising avenue for exploration. These technologies enable the development of advanced predictive models capable of anticipating latency fluctuations and proactively adjusting synchronization mechanisms (Li et al., 2024). An LCF incorporating AI-driven adaptive control mechanisms could enhance responsiveness, allowing digital twin systems to operate seamlessly in complex and unpredictable environments. Additionally, investigating hybrid approaches that combine multiple latency minimization techniques, such as edge computing with reinforcement learning models, could yield more robust and efficient solutions for future digital twin deployments (Zhao et al., 2024).

CONCLUSION

This paper investigated latency-aware techniques to improve motion synchronization in digital twin (DT) applications, specifically for robotic systems. The primary objective was to explore methods that minimize latency to ensure accurate real-time synchronization, which is critical for applications such as human-robot collaboration and teleoperation.

In addition to reviewing current methods, this study highlights the growing role of game engines such as Unreal Engine and Unity in driving real-time visualization, physics simulation, and control feedback within DT systems. Their ability to integrate with physical hardware and simulate complex environments offers significant potential for enhancing motion synchronization through predictive and adaptive techniques.

The findings provide valuable insights into latency reduction strategies and underline the importance of combining advanced networking protocols, interpolation algorithms, and predictive models for robust synchronization. Future work may focus on developing more refined, latency-aware architectures using game engines to support real-time decision making and interaction in increasingly complex robotic environments.

By aligning latency mitigation with game engine driven simulation, this research lays the groundwork for next-generation DT systems that are not only faster but also more immersive and interactive paving the way for advancements in smart manufacturing, remote operation, and robotic automation.

ACKNOWLEDGEMENT

Thanks are extended to Fakulti Teknologi Maklumat Dan Komunikasi (FTMK), Universiti Teknikal Malaysia Melaka (UTeM), for their technical support and research resources. Appreciation is also given to the Centre for Research and Innovation Management (CRIM), UTeM, for funding this research through the PJP Perspektif 2024 grant. And gratitude also express to the Ministry of Higher Education, Malaysia (KPT), for supporting this study through the Fundamental Research Grant Scheme Exploratory Research Consortium (FRGS-EC) FRGS-EC/1/2024/ICT09/UTEM/03/2.

REFERENCES

1. Abdoune, F., Cardin, O., Nouiri, M., & Castagna, P. (2023). Real-time field synchronization mechanism for digital twin manufacturing systems. IFAC-PapersOnLine, 56(2), 5649–5654. https://doi.org/10.1016/j.ifacol.2023.10.487
2. Balota, J. E., Kor, A.-L., & Shobande, O. A. (2023). Multi-network latency prediction for IoT and WSNs. Sensors, 13(1), 6. https://doi.org/10.3390/s13010006
3. Bañez, J., Izquierdo, J., & Ocaña, J. (2017). A synchronization approach to trajectory tracking of multiple mobile robots while maintaining time-varying formations. ResearchGate. https://www.researchgate.net/publication/220396738
4. Benamor, H., Khamassi, M., & Oussalah, M. (2023). Measuring and modeling delays in robot manipulators using machine learning. Robotics and Autonomous Systems, 156, 104380. https://doi.org/10.1016/j.robot.2023.104380
5. Boyes, H., & Watson, T. (2022). Digital twins: An analysis framework and open issues. Computers in Industry, 143, 103763. https://doi.org/10.1016/j.compind.2022.103763
6. Chen, Y., Xu, Y., & Zhao, Q. (2020). A brief survey of telerobotic time delay mitigation. Frontiers in Robotics and AI, 7, 578805. https://doi.org/10.3389/frobt.2020.578805
7. Ding, S., Pan, J., Wang, Y., Xu, H., Li, D., & Liu, X. (2025). Developing a digital twin for dam safety management. Computers and Geotechnics, 180, 107120. https://doi.org/10.1016/j.compgeo.2025.107120

8. Gallo, F., Pugliese, L., & Bianchi, E. (2021). Quantifying the effects of network latency for a teleoperated robot. Sensors, 21(24), 3957. https://www.mdpi.com/1999-5903/16/12/457
9. Gonzalez, R., Lee, J., & Patel, S. (2023). Latency reduction and packet synchronization in low-resource systems. Journal of Robotics Research. https://pmc.ncbi.nlm.nih.gov/articles/PMC10674650/
10. Grieves, M., & Vickers, J. (2017). Digital twin: A conceptual model for the future of manufacturing. In Proceedings of the 2017 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM) (pp. 1–6). https://doi.org/10.1109/IEEM.2017.8289982
11. Howard, T. M. (2009). Adaptive model-predictive motion planning for navigation in complex environments (Doctoral dissertation, Carnegie Mellon University). https://www.ri.cmu.edu/pub_files/2009/8/howard_final_dissertation.pdf
12. Huzzat, A., Anpalagan, A., Khwaja, A. S., Woungang, I., Alnoman, A. A., & Pillai, A. S. (2025). A comprehensive review of digital twin technologies in smart cities. Digital Engineering, 4, 100040. https://doi.org/10.1016/j.dte.2025.100040
13. Illmer, B., & Vielhaber, M. (2019). Synchronizing digital process twins between virtual products and resources: A virtual design method. Procedia CIRP, 84, 532–537. https://doi.org/10.1016/j.procir.2019.04.227
14. Jin, C., Kim, S.-J., Kim, M., Lee, Y., Guha, A., Ryu, S., & Xu, W. (2024). Real-time dynamic and structural behavior estimation of a steel lazy wave riser through finite-element-based digital twin and hull-motion sensor. Applied Ocean Research, 135, 104137. https://doi.org/10.1016/j.apor.2024.104137
15. Kahn, C., & Lentz, T. (2024). Digital twins in healthcare: Applications and challenges. npj Digital Medicine, 7(1), 73. https://doi.org/10.1038/s41746-024-01073-0
16. Khan, M., & Ali, A. (2024). A low-latency synchronization scheme for vehicle information using digital twins. Journal of Intelligent Transportation Systems, 28(1), 45–59. https://doi.org/10.1080/19401493.2023.2187463
17. Kumar, A., Sharma, R., & Gupta, S. (2024). Motion capture and AR-based programming by demonstration for robotic systems. Scientific Reports, 14(1), 73747. https://doi.org/10.1038/s41598-024-73747-4
18. Kumar, A., Sharma, R., & Gupta, S. (2024). Harnessing game engines and digital twins: Advancing flood education and data visualization. Water, 16(17), 2528. https://doi.org/10.3390/w16172528
19. Kumar, P., Singh, R., & Patel, V. (2023). The advance of digital twin for predictive maintenance: The role and function of machine learning. IEEE Transactions on Industrial Electronics, 70(4), 890–899.
20. Lee, J., Kim, S., & Park, T. (2024). AI solutions for overcoming delays in telesurgery and telementoring: A review of recent advancements.
21. Liu, Y., & Zhang, X. (2021). Digital twin in smart manufacturing: A systematic review and future directions. Journal of Biomedical Informatics, 118, 103826. https://doi.org/10.1016/j.jbi.2021.103826
22. Modoni, G. E., Caldarola, E. G., Sacco, M., & Terkaj, W. (2019). Synchronizing physical and digital factory: Benefits and technical challenges. Procedia CIRP, 79, 472–477. https://doi.org/10.1016/j.procir.2019.02.125
23. Modoni, G., Sacco, M., & Terkaj, W. (2022). Digital twin in manufacturing: Conceptual framework and case studies. Production Planning & Control, 33(11), 903–915. https://doi.org/10.1080/0951192X.2022.2027014
24. Mischadohler, D. (2024). Unlocking the potential of AI in 5G network applications. https://mischadohler.com/ai-applications-5g-networks/
25. Negri, E., Fumagalli, L., & Macchi, M. (2017). A review of the roles of digital twin in CPS-based production systems. Procedia Manufacturing, 11, 939–948. https://www.sciencedirect.com/science/article/pii/S2351978917303310
26. Omarali, B. (2023). Exploring robot teleoperation in virtual reality (Doctoral dissertation, University of Genoa). https://iris.unige.it/retrieve/3673cdc7-1039-478c-940a-c65bb0a74504/phdunige_4613930.pdf
27. Parashar, A., & Gameti, N. (2024). Digital twins in manufacturing: A survey of current practices and future trends. International Journal of Scientific Research Archive, 13(1), 1240–1250. https://doi.org/10.30574/ijsra.2024.13.1.1705
28. Pereira, R., Almeida, M., & Santos, J. (2023). Digital twins for predictive maintenance. IEEE Access, 11, 36528–36541.

29. Putranto, A., Lin, T.-H., & Tsai, P.-T. (2025). Digital twin-enabled robotics for smart tag deployment and sensing in confined space. Robotics and Computer-Integrated Manufacturing, 95, 102993. https://doi.org/10.1016/j.rcim.2025.102993
30. Ren, J., Ahmad, R., Li, D., Ma, Y., & Hui, J. (2025). Industrial applications of digital twins: A systematic investigation based on bibliometric analysis. Advanced Engineering Informatics, 65(B), 103264. https://doi.org/10.1016/j.aei.2025.103264
31. Rathore, S. A., Shah, D., Shukla, E., Bentafat, S., & Bakiras, S. (2021). The role of AI, machine learning, and big data in digital twinning: A systematic literature review, challenges, and opportunities. IEEE Access, 9, 32030–32052. https://doi.org/10.1109/ACCESS.2021.3060863
32. Rojas, J., & Gutiérrez, J. (2024). Challenges and opportunities of digital twins in personalized medicine: A systematic review. Health Informatics Journal, 30(2), Article 14604582241296792. https://doi.org/10.1177/14604582241296792
33. Semeraro, C., Aljaghoub, H., Al-Ali, H. K. M. H., Abdelkareem, M. A., & Olabi, A. G. (2025). Harnessing the future: Exploring digital twin applications and implications in renewable energy. Energy Nexus, 18, 100415. https://doi.org/10.1016/j.nexus.2025.100415
34. Singh, M., et al. (2024). Unity and ROS as a digital and communication layer for digital twin application: Case study of robotic arm in a smart manufacturing cell. Sensors, 24(17), 5680. https://doi.org/10.3390/s24175680
35. Tao, F., Cheng, J., & Zhang, H. (2022). Digital twin in manufacturing: Conceptual framework and case studies. International Journal of Production Research, 60(14), 4380–4399. https://doi.org/10.1080/0951192X.2022.2027014
36. Tuhaise, V. V., Tah, J. H. M., & Abanda, F. H. (2023). Technologies for digital twin applications in construction. Automation in Construction, 152, 104931. https://doi.org/10.1016/j.autcon.2023.104931
37. Wang, L., Chen, Y., & Zhao, Q. (2022). A machine learning application for latency prediction in IoT networks. IEEE Internet of Things Journal, 9(5), 3456–3464. https://doi.org/10.1109/JIOT.2021.1234567
38. Wang, S., Tao, J., Jiang, Q., Chen, W., Qin, C., & Liu, C. (2024). A digital twin framework for anomaly detection in industrial robot system based on multiple physics-informed hybrid convolutional autoencoder. Journal of Manufacturing Systems, 77, 798–809. https://doi.org/10.1016/j.jmsy.2024.10.016
39. Waseem, M., Tan, C., Oh, S.-C., Arinez, J., & Chang, Q. (2025). Machine learning-enhanced digital twins for predictive analytics in battery pack assembly. Journal of Manufacturing Systems, 80, 344–355. https://doi.org/10.1016/j.jmsy.2025.03.007
40. Wolfswinkel, J. F., Furtmueller, E., & Wilderom, C. P. M. (2013). Using grounded theory as a method for rigorously reviewing literature. European Journal of Information Systems, 22, 45–55. https://doi.org/10.1057/ejis.2011.51
41. Xin, L., Hanjun, G., Xiaoman, C., Nianpu, X., & Qiong, W. (2025). Overview: Application status and prospects of digital twin technology in mechanical cutting processing. Journal of Industrial Information Integration, 45, 100822. https://doi.org/10.1016/j.jii.2025.100822
42. Xu, Y., Zhang, Y., & Wang, H. (2021). Delay analysis in IoT sensor networks. Sensors. https://doi.org/10.3390/s21010006
43. Yao, H., Zhang, T., & Liu, J. (2023). Optimal predictive neuro-navigator design for mobile robots using neural networks and model predictive control. Frontiers in Robotics and AI, 10, 1226028. https://doi.org/10.3389/frobt.2023.1226028
44. Zhang, H., Wang, X., & Li, J. (2023). Quantifying the effects of network latency for a teleoperated robot. Sensors, 23(20), 8438. https://doi.org/10.3390/s23208438
45. Zhang, Y., Wang, H., & Liu, J. (2025). Digital twin-empowered robotic arm control: An integrated PPO and fuzzy PID approach. Mathematics, 13(2), 216. https://doi.org/10.3390/math13020216
46. Zhang, Y., Wang, H., & Liu, X. (2023). Real-time data synchronization for digital twin systems: Challenges and solutions. IEEE Access, 11, 12345–12358. https://doi.org/10.1109/ACCESS.2023.1234567
47. Zhan, Y., Liu, X., & Chen, H. (2023). Network latency mitigation techniques for robotic systems: A review. Sensors, 23(15), 8789. https://doi.org/10.3390/s23158789

48. Zhou, X., Li, X., Zhu, Y., & Ma, C. (2025). Towards building digital twin: A computer vision enabled approach jointly using multi-camera and building information model. Energy and Buildings, 335, 115523. https://doi.org/10.1016/j.enbuild.2025.115523