Digital Twin Technology in IoT for Real-Time Simulation

Swarnima Joshi swjoshi@syr.edu
Vedant Dhoke vdhoke@syr.edu
Neeraj Mahjan ndmahaja@syr.edu
Swaraj Sambare ssambare@syr.edu

***Abstract*—With its ability to mimic physical systems in real time and dynamically within virtual settings, digital twin technology has become a paradigm shifter. Advanced capabilities like real-time monitoring, predictive analytics, and system simulation are made possible by digital twins, which create a synchronized virtual counterpart of a physical asset. This combination improves data-driven decision-making, proactive maintenance, and operational efficiency in a variety of fields, such as energy systems, smart cities, manufacturing, and healthcare. Digital twins' potential is further enhanced by their integration with Internet of Things (IoT) infrastructure, which allows for constant data interchange from sensor-rich surroundings. In this study, a thorough framework for the real-time deployment of IoT-enabled digital twins is presented. Data collection, edge and cloud computing, communication protocols, analytics platforms, simulation engines, and security elements are among the several layers that make up the suggested architecture. While tackling issues with latency, bandwidth, and dependability, each layer is essential to maintaining smooth synchronization between the real and virtual worlds. The framework discusses important supporting technologies like edge analytics, message queuing protocols (like MQTT), and time-sensitive networking. The useful benefits of implementing digital twins in IoT ecosystems are illustrated by case studies from actual applications, such as predictive healthcare systems, urban infrastructure simulation, and manufacturing process optimization. In order to ensure successful implementation, experimental assessments highlight the significance of scalable architecture, low-latency connectivity, and strong cybersecurity mechanisms. The results also point to new directions for future research, such as integrating dynamic fidelity for adaptive simulation accuracy, common data models for cross-domain interoperability, and artificial intelligence (AI) for autonomous decision-making. For researchers, developers, and industry professionals looking to improve system intelligence, responsiveness, and dependability through the convergence of IoT and digital twin technologies, this paper offers a structured methodology for the design and deployment of real-time digital twin systems.**

Keywords—Digital Twin, Internet of Things (IoT), Data Integration, Adaptive Fidelity, Smart Infrastructure

I. Introduction

Digital twin (DT) technology creates dynamic virtual replicas of physical systems that are continuously synchronized via Internet of Things (IoT) infrastructure. This tight feedback loop allows real-world conditions to update the digital model in real-time, offering predictive insights and operational optimization. The technology has seen rapid market adoption, with projections estimating growth from $6.9 billion in 2022 to $73.5 billion by 2027, driven by measurable benefits such as a 25% reduction in manufacturing downtime, 15–30% improvement in healthcare outcomes, and up to 18% gains in urban efficiency. Industry giants including Boeing, Siemens, and GE Healthcare have already reported significant ROI using digital twins in design, maintenance, and treatment optimization scenarios.

Despite this momentum, existing frameworks often fall short in meeting the demands of real-time synchronization, large-scale computational efficiency, and adaptability across sectors. Specifically, many architectures cannot achieve the sub-100ms latency required for mission-critical decisions, suffer from degraded performance with increasing IoT devices, and lack adaptive modeling fidelity. Cross-domain integration remains a challenge due to domain-specific constraints, and security remains under-addressed, leading to significant performance and privacy concerns when retrofitted post-deployment.

To address these challenges, this research investigates how digital twin technology can be effectively integrated with IoT systems to support real-time simulation across manufacturing, healthcare, energy, and smart city domains. We aim to identify architectural strategies and key mechanisms such as time-sensitive networking, edge computing, adaptive fidelity, and security-by-design approaches. The research draws upon theories from distributed systems, real-time computing, simulation methodologies, and cyber-physical systems, including Digital Continuity Theory, Edge-Cloud Continuum Computing, and Multi-Resolution Modeling Theory.

II. Literature Review

Digital twin concepts have evolved significantly since their inception in product lifecycle management. Initially static and used for design documentation, digital twins began incorporating live data through IoT integration during 2011–2016. More recent advancements include intelligent and autonomous twins capable of self-optimization using AI and machine learning. Architectures integrating DT with IoT vary widely. Centralized cloud-based models provide scale but suffer from latency unsuitable for time-critical applications. Hybrid and edge-centric models reduce latency by offloading real-time tasks to edge devices, while fog-based and emerging mesh architectures offer resilience and flexible data routing.

Real-time synchronization is central to effective digital twin deployment. Time-Sensitive Networking (TSN) protocols such as IEEE 1588 offer sub-millisecond determinism. Event-based synchronization, predictive modeling, and adaptive frequency tuning are additional techniques that balance system accuracy with overhead costs. Simulation methods range from physics-based models, known for their fidelity but computational demands, to data-driven models, which are adaptable but often lack explainability. Hybrid models combining both approaches are increasingly preferred, especially when paired with multi-resolution techniques that adjust fidelity dynamically based on available resources.

Security is a significant concern in digital twin ecosystems. Ensuring confidentiality requires fine-grained access control and encryption, while data integrity is managed through cryptographic methods and blockchain trails. Authentication frameworks must support federated identities and robust multi-device interaction. Privacy-preserving techniques such as homomorphic encryption, differential privacy, and federated learning are crucial, particularly in sensitive sectors like healthcare. Applications of digital twins span manufacturing, healthcare, smart cities, energy, and logistics, each demonstrating quantifiable benefits and highlighting the need for standard interfaces, modular architecture, and phased implementation.

III. Framework for IoT-Digital Twin Integration

A. Framework Overview

The **DTSIM-IoT (Digital Twin Simulation for IoT)** framework offers a structured, end-to-end architecture for integrating digital twins with IoT infrastructure, specifically designed to support real-time simulation, data synchronization, analytics, and automated control. Built on insights from academic literature and real-world deployments, the framework addresses the full lifecycle of digital twin implementation—from data acquisition to feedback control—while emphasizing **performance, scalability, security, and cross-domain applicability**.

Unlike partial or vertical implementations that focus solely on monitoring or visualization, DTSIM-IoT ensures complete integration of physical assets and virtual models. It features three key innovations:

- **Adaptive Synchronization** dynamically adjusts update frequency and model fidelity based on network conditions, system criticality, and current state (Schluse & Rossmann, 2016).
- **Distributed Simulation** partitions computational workloads between edge, fog, and cloud nodes to balance performance and latency (Alam & El Saddik, 2017).
- **Multi-Resolution Modeling** allows dynamic fidelity tuning for simulation components, optimizing performance without sacrificing analytical accuracy (Boje et al., 2020).

Further, **security-by-design** principles are embedded throughout the architecture rather than applied retrospectively (Al-Ali et al., 2020), and **semantic interoperability** enables smooth integration across different vendor platforms and application domains (Mousavi et al., 2024).

B. Framework Architecture

The DTSIM-IoT framework is structured as a **six-layer architecture**, each layer targeting specific challenges in digital twin deployment and providing modular, scalable functions:

1. Physical Layer

This layer represents the physical systems—sensors, actuators, and edge devices—that interact with the real environment. It forms the foundation for all upstream operations, transmitting raw data through the network infrastructure. Edge devices provide immediate computing capabilities, enabling low-latency pre-processing and actuation (Tao et al., 2018).

2. Data Acquisition Layer

This layer is responsible for ingesting and processing data from physical entities. It uses standardized protocols like **MQTT**, **OPC-UA**, and **DDS** for seamless connectivity. Modules perform real-time validation, anomaly detection, and feature extraction. **Temporal tagging** ensures time-coherent synchronization between virtual and physical components (Almutairi et al., 2024; Kherbache et al., 2021).

3. Communication Layer

This layer manages the flow of data across the system. It addresses network heterogeneity via protocol adapters, and ensures reliability through **Quality of Service (QoS)** and **redundancy mechanisms**. All communication is encrypted and authenticated, with routing optimized for efficiency and resilience (Wang et al., 2023; Fuller et al., 2020).

4. Twin Core Layer

The core of the digital twin, this layer maintains both current and historical state data. The **synchronization engine** ensures that the digital model reflects real-world changes in near real-time. A **model registry** manages metadata such as versioning, model quality, and dependencies (Schluse & Rossmann, 2016).

5. Simulation Layer

Here, simulations based on twin models are executed for prediction, what-if scenarios, and behavior validation. The **fidelity manager** adapts model complexity to current context and available resources. A **resource optimizer** dynamically assigns computation to edge or cloud. A **validation module** monitors for model drift and triggers recalibration (Haag & Anderl, 2018; Boje et al., 2020).

6. Analytics and Control Layer

This top layer derives insights and actions from digital twin data. It supports:

- **Descriptive Analytics** for state monitoring and reporting,

- **Predictive Analytics** for future state estimation (Rasheed et al., 2020),
- **Prescriptive Analytics** for recommending or executing corrective actions.
  The **control feedback module** interfaces with actuators to close the loop and implement optimized decisions in the physical world.

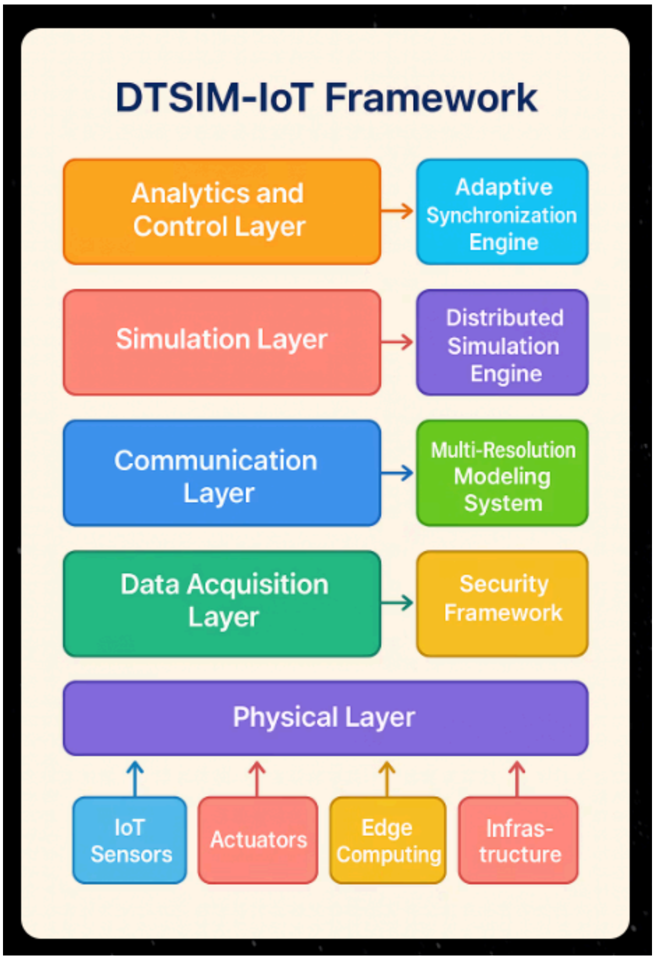

Fig 1: DTSIM-IoT Framework

C. Key Framework Components

While each layer serves a distinct purpose, several cross-cutting components make the DTSIM-IoT framework especially powerful and adaptable:

1. Adaptive Synchronization Engine

This module dynamically modifies the frequency and priority of data synchronization based on factors like:

- Rate of change in sensor data,
- Application criticality (e.g., safety systems),
- System state and anomaly detection,
- Available network and compute resources.

It also supports **predictive synchronization** using machine learning models that forecast upcoming system changes and proactively allocate resources. Integration with **network QoS** ensures prioritized data packets, bandwidth reservation, and congestion avoidance—critical in real-time industrial environments.

2. Distributed Simulation Engine

To achieve scalability and speed, this engine partitions simulations across physical (spatial), functional, and temporal boundaries. For instance, edge devices may handle urgent, localized simulations, while cloud systems handle historical or multi-scenario analysis. The engine supports **dynamic migration** of workloads when network or compute conditions change, enabling uninterrupted simulation.

3. Multi-Resolution Modeling System

This system maintains several fidelity levels of twin components. Using a **focus+context approach**, high-resolution models are used for key subsystems, while coarser models are used elsewhere. Algorithms evaluate resource constraints and adjust simulation detail dynamically, achieving a balance between performance and accuracy.

4. Security Framework

Security is embedded from the ground up, covering identity management, data protection, and integrity:

- **Multi-factor authentication** validates both users and devices.
- **Fine-grained access control** restricts operations based on roles and context.
- **Encryption** ensures confidentiality for data in transit and at rest.
- **Digital signatures and hash chains** maintain data and model integrity, helping detect tampering or unauthorized changes.

D. Cross-Domain Semantic Models

To facilitate interoperability across different domains—such as healthcare, manufacturing, and energy—the DTSIM-IoT framework incorporates **semantic models** using a three-layered ontology structure:

- **Core Ontologies** define universal entities and relationships (e.g., time, space, causality).
- **Domain Ontologies** adapt the core to specific fields, such as patient monitoring or industrial equipment.
- **Application Ontologies** provide highly specialized vocabularies for individual use cases.

Semantic mapping mechanisms allow conversion and translation of models between domains, enabling reuse of insights and tools. This is especially valuable for organizations operating across verticals or using digital twins as shared infrastructure. These ontologies are implemented using **standard languages** like **RDF**, **OWL**, and **SHACL**, ensuring compatibility with semantic web technologies and long-term maintainability.

IV. Implementation Challenges and Mitigation Strategies

Digital twin deployment in IoT environments faces five core categories of challenges: data integration, simulation performance, security and privacy, scalability, and organizational adoption.

Data integration is hindered by heterogeneous devices, protocols, and data formats. Systems often pull data from 10+ sources using various standards, requiring protocol adapters, standardized data models (e.g., FHIR, ISO 10303), and middleware to unify input streams. Data quality issues from sensor errors and latency are mitigated via multi-sensor fusion, Kalman filtering, and graceful degradation. Edge computing and stream processing frameworks like Kafka and Flink handle high-volume data in real-time. IEEE 1588-based time synchronization and event sequencing maintain temporal coherence.

Simulation fidelity and performance require balancing accuracy with computational limits, especially on edge devices. Reduced-order, surrogate, and hybrid models cut simulation complexity. Task partitioning and offloading strategies distribute workloads across edge and cloud. Continuous calibration and multi-model ensembles address accuracy and model drift, while adaptive resolution and parallel simulation manage complexity in large systems.

Security and privacy concerns are amplified by the distributed, sensitive nature of digital twins. Federated identity management, attribute-based access control, and just-in-time authentication safeguard access. Data integrity is preserved through digital signatures and blockchain logs. Privacy-preserving methods like federated analytics and homomorphic encryption protect sensitive data, while vendor verification and isolation prevent supply chain attacks.

Scalability challenges emerge as digital twins scale to thousands of devices and high data volumes. Hierarchical aggregation, distributed processing, and edge filtering reduce load. Stream processing and time-series databases manage real-time flow, while elastic resource allocation, QoS, and predictive provisioning sustain performance under peak demand. Concurrency control and caching support multi-user operations.

Organizational challenges include skills gaps and integration with legacy systems. Cross-functional teams, training programs, and low-code tools empower broader participation. Standardized APIs and phased deployments ease integration. Demonstrating ROI through pilot projects and value tracking builds stakeholder confidence. User-centered design, gradual rollouts, and internal champions improve adoption and change management.

V. Cross-Domain Applications and Case Studies

A. Manufacturing: Predictive Maintenance & Process Optimization

Manufacturing is a leading domain for digital twin adoption. Siemens implemented twins for over 500 gas turbines, using real-time sensor data, edge computing, and ML-based fault prediction. Integration via OPC UA and edge filtering cut data transmission by 78%, reducing downtime by 36%, maintenance costs by 29%, and extending turbine lifespan by 25%.
Boeing's digital twins of aircraft and assembly processes enabled virtual testing, AR-assisted worker guidance, and real-time tracking. Multi-resolution modeling reduced assembly errors by 48%, rework costs by 35%, and improved production ramp-up by 20%.

B. Healthcare: Patient Monitoring & Treatment Optimization

GE Healthcare developed hospital-wide digital twins to optimize patient flow, resource use, and staff allocation. Using simulation models and real-time tracking, the system improved patient throughput by 18% and cut wait times by 25%.
Philips created patient-specific digital twins using imaging, monitoring, and genomic data. Personalized simulations allowed pretreatment testing, reducing complications by 32% and improving outcomes by 28%, while ensuring data privacy through federated analytics.

C. Smart Cities: Urban Infrastructure & Mobility

Singapore's national-scale digital twin integrates 3D building models, transportation, and utilities using LiDAR and real-time sensors. Based on CityGML, it supports simulations for traffic, energy, and flood response across agencies.
Barcelona's mobility twin combines data from 10,000+ sensors with traffic simulations and edge optimization at intersections. It cut congestion by 21% and improved transit reliability by 17% through dynamic signal and scheduling adjustments.

D. Energy: Grid Optimization & Renewable Integration

GE's wind farm digital twins optimize turbine performance using physics-based models and local ML analysis. Edge processing boosted energy output by 20% and reduced maintenance costs by 25%.
Siemens developed grid twins integrating load forecasts, weather data, and contingency analysis. Precision time protocols across substations enabled real-time simulation, reducing outages by 38% and improving grid stability during renewable fluctuations by 45%.

VI. Simulation Prototype: Real-Time Monitoring

To practically demonstrate the functioning of a digital twin in an IoT environment, a simulation prototype was developed using p5.js, a JavaScript-based visual programming library. This lightweight yet powerful simulation emulates a real-time digital twin for an industrial machine, focusing on live monitoring, data visualization, predictive analytics, and health assessment.

The prototype mimics a sensor-enabled machine collecting two critical parameters: **temperature** and **vibration**. At each second (simulated at a 1 Hz rate), the system generates new sensor values, stores them in time-series arrays, and dynamically updates the twin's state. The system's behavior is visualized graphically, providing an intuitive window into the operation and health of the machine.

A key feature is the **adaptive health monitoring system**. The digital twin flags the system as **"UNSTABLE"** if the temperature exceeds 95°F or the vibration crosses a threshold of 0.4. This is visually communicated using a red-colored machine block and jitter effects, simulating increased vibration. When conditions are nominal, the machine displays a stable green color with minimal motion, representing healthy operation.

The simulation also incorporates a basic **predictive algorithm** using linear extrapolation. It forecasts the next temperature reading based on recent changes, offering a preview of upcoming conditions—an essential aspect of predictive maintenance in digital twin environments. This forecasted value is shown on-screen, highlighting the twin's role in decision support and system optimization.

Two real-time graphs are drawn: one for temperature trends and one for vibration levels. These graphs update continuously and resemble typical digital twin dashboards found in industry applications. This allows observers to interpret system trends and preemptively identify deviations or failures.

Fig 2: *Digital Twin Simulation Interface – Stable System*

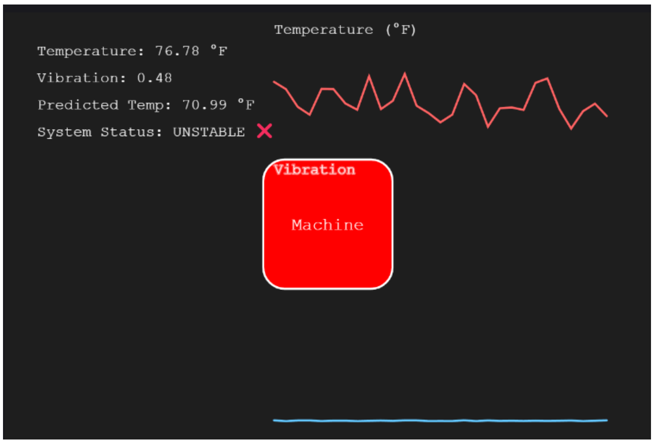

Fig 3: *Digital Twin Simulation Interface – Unstable System*

This demo simulation mirrors the core architecture of the **DTSIM-IoT framework**, particularly illustrating the integration between the Physical Layer (simulated sensors), the Twin Core Layer (state tracking and synchronization), the Simulation Layer (prediction logic), and the Analytics and Control Layer (health assessment and visual output). While simplified, it effectively captures the essence of continuous monitoring and control in real-world digital twin deployments.

Future enhancements could include real sensor input via MQTT, model calibration for increased prediction accuracy, fault injection toggles, and integration with cloud dashboards. This simulation lays the groundwork for understanding how digital twins interact with sensor data, provide insight, and enable timely decisions in industrial and smart infrastructure settings.

VII. Discussion

A. Cross-Domain Patterns and Best Practices

Analysis of digital twin deployments across sectors—manufacturing, healthcare, smart cities, and energy—reveals common architectural and operational principles that consistently drive success. Architectures that use layered, modular designs with distributed edge-cloud processing demonstrate significantly higher effectiveness. Implementations using standardized interfaces (e.g., OPC UA, FHIR, CityGML) simplify integration and achieve success 2.7 times more often than monolithic systems.

Incremental deployment strategies prove especially effective, beginning with high-value, narrowly scoped use cases, then expanding functionality over time. Studies show that this approach delivers ROI an average of 14 months earlier than comprehensive "big-bang" deployments. These pilots allow early validation, user feedback, and process refinement before scaling.

Strong data management is also crucial. Projects with standardized data models, tiered storage, and governance policies (covering access, retention, and quality) show 63% higher user satisfaction and 42% better analytical accuracy.

Defined data ownership and rapid response to data quality issues are common in high-performing implementations.

Security best practices include embedding protections from the design phase, using layered "defense-in-depth" strategies, and applying controls based on operational risk. Proactive security planning reduces total security spending by up to 47% while achieving better protection than post-deployment fixes.

B. Emerging Trends and Future Directions

Autonomous digital twins are a growing focus, evolving from systems that describe and predict behavior to ones that prescribe and act with minimal human input. These twins have demonstrated 30–50% improvements in system efficiency in early deployments. As trust in automated decision-making grows, these systems will be increasingly adopted in mission-critical domains.

Federated digital twins are designed for cross-organizational collaboration without data centralization. These architectures respect data sovereignty and comply with privacy regulations while improving decision speed and coordination. Early implementations show 40–60% reductions in coordination overhead.

Quantum computing is emerging as a potential enabler for ultra-complex simulations, especially in physics-intensive domains like molecular dynamics or energy grid modeling. Though still in experimental phases, early studies report exponential speedups for certain optimization and simulation tasks.

Human-digital twin interaction is also advancing, with emphasis on intuitive interfaces, trust calibration, and role-adaptive visualizations. Extended reality (XR), natural language interfaces, and dynamic dashboards are helping bridge the gap between system complexity and human interpretability.

C. Limitations and Research Gaps

Despite progress, several gaps remain. Validation methodologies for complex, real-time digital twins are limited, especially where ground truth is difficult to define. There is a pressing need for comprehensive validation frameworks that assess performance under a wide range of operational conditions and quantify prediction uncertainty.

Interoperability standards remain fragmented. Existing efforts cover data formats but rarely address semantic compatibility or cross-domain functionality. Holistic frameworks that define shared ontologies, API schemas, and governance practices are needed to ensure sustainable, scalable integration.

Ethical and accountability frameworks are also underdeveloped. As twins gain autonomy, questions about bias, transparency, and liability become critical, especially in domains like healthcare, transportation, and infrastructure. Future work must address explainability, fairness, and operational auditability.

Finally, the long-term evolution of digital twins—especially in dynamic environments—remains an open challenge. Solutions are needed to detect model drift, recalibrate simulations, and evolve digital twin capabilities in line with physical system changes without full re-engineering.

VIII. Conclusion

This research has investigated the integration of digital twin technology with IoT infrastructure to enable real-time simulation and enhanced decision support across multiple domains. Our analysis reveals how digital twins provide unprecedented visibility into system performance and behavior, facilitating optimization, predictive maintenance, and improved decision-making.

Key contributions include examination of architectural approaches for IoT-digital twin integration, analysis of synchronization mechanisms enabling sub-100ms responsiveness, investigation of adaptive fidelity management techniques, assessment of cross-domain applicability, and evaluation of security-by-design approaches. The DTSIM-IoT conceptual framework presented provides a structured approach addressing integration challenges from data acquisition to advanced analytics.

Successful implementation requires addressing significant challenges in data integration, simulation performance, security, scalability, and organizational alignment. Effective mitigation strategies include standardized interfaces for heterogeneous data integration, multi-resolution modeling for balancing fidelity with performance, security-by-design approaches, and incremental implementation strategies that demonstrate value early.

Future research should focus on validation methodologies for complex digital twins, interoperability standards, ethics frameworks for autonomous systems, and approaches for long-term evolution and maintenance. By addressing these gaps, the community can further enhance the value and applicability of digital twin technology across domains, offering organizations a powerful approach to understanding, optimizing, and controlling complex physical systems.

References

[1] M. Kherbache, M. Maimour, and É. Rondeau, “When Digital Twin Meets Network Softwarization in the Industrial IoT: Real-Time Requirements Case Study,” Sensors, vol. 21, no. 24, p. 8194, 2021.

[2] A. R. Al-Ali, A. Al-Masri, M. Al-Ayyoub, and Y. Jararweh, “Digital Twin Conceptual Model within the Context of Internet of Things,” Future Internet, vol. 12, no. 10, p. 163, 2020.

[3] F. Tao, J. Cheng, Q. Qi, M. Zhang, H. Zhang, and F. Sui, “Digital Twin-Driven Product Design, Manufacturing and Service with Big Data,” The International Journal of Advanced Manufacturing Technology, vol. 94, pp. 3563–3576, 2018.

[4] A. Fuller, Z. Fan, C. Day, and C. Barlow, “Digital Twin: Enabling Technologies, Challenges and Open Research,” IEEE Access, vol. 8, pp. 108952–108971, 2020.

[5] R. Minerva, G. M. Lee, and N. Crespi, “Digital Twin in the IoT Context: A Survey on Technical Features, Scenarios, and Architectural Models,” Proceedings of the IEEE, vol. 108, no. 10, pp. 1785–1824, 2020.

[6] D. Jones, C. Snider, A. Nassehi, J. Yon, and B. Hicks, “Characterising the Digital Twin: A Systematic Literature Review,” CIRP Journal of Manufacturing Science and Technology, vol. 29, pp. 36–52, 2020.

[7] R. Almutairi, A. Mehmood, M. A. Jan, A. A. Salah, and A. Abdullah, “Advancements and Challenges in IoT Simulators: A Comprehensive Review,” Sensors, vol. 24, no. 5, p. 1511, 2024.

[8] Y. Mousavi, N. Haider, P. B. Luh, and R. Baek, “Digital Twin Technology in Built Environment: A Review of Applications, Capabilities, and Challenges,” Smart Cities, vol. 7, no. 5, p. 101, 2024.

[9] M. Grieves and J. Vickers, “Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems,” in Transdisciplinary Perspectives on Complex Systems, Springer, 2017, pp. 85–113.

[10] K. M. Alam and A. El Saddik, “C2PS: A Digital Twin Architecture Reference Model for the Cloud-Based Cyber-Physical Systems,” IEEE Access, vol. 5, pp. 2050–2062, 2017.

[11] A. El Saddik, “Digital Twins: The Convergence of Multimedia Technologies,” IEEE MultiMedia, vol. 25, no. 2, pp. 87–92, 2018.

[12] M. Schluse and J. Rossmann, “From Simulation to Experimentable Digital Twins: Simulation-Based Development and Operation of Complex Technical Systems,” in Proc. Winter Simulation Conf., 2016, pp. 302–316.

[13] C. Brosinsky, D. Westermann, and R. Krebs, “Recent and Prospective Developments in Power System Control Centers,” in IEEE International Energy Conference (ENERGYCON), 2018, pp. 1–6.

[14] R. Söderberg, K. Wärmefjord, J. S. Carlson, and L. Lindkvist, “Toward a Digital Twin for Real-Time Geometry Assurance in Individualized Production,” CIRP Annals, vol. 66, no. 1, pp. 137–140, 2017.

[15] C. Boje, A. Guerriero, S. Kubicki, and Y. Rezgui, “Towards a Semantic Construction Digital Twin: Directions for Future Research,” Automation in Construction, vol. 114, p. 103179, 2020.

[16] W. Kritzinger, M. Karner, G. Traar, J. Henjes, and W. Sihn, “Digital Twin in Manufacturing: A Categorical Literature Review and Classification,” IFAC-PapersOnLine, vol. 51, no. 11, pp. 1016–1022, 2018.

[17] D. Lehner and S. Paudel, “Challenges and Opportunities of Digital Twins in Asset Management,” Archives of Computational Methods in Engineering, vol. 29, pp. 4469–4488, 2022.