Distributed Edge-Fog-Cloud Framework for Hierarchical Digital Twins

1st Erol ÖZKAN
Department of Computer Engineering
Hacettepe University
Ankara, Turkey
erolozkan@outlook.com

2rd Suat ÖZDEMİR
Department of Computer Engineering
Hacettepe University
Ankara, Turkey
ozdemir@cs.hacettepe.edu.tr

***Abstract*—Digital Twin (DT) technology creates real-time virtual models of physical systems, enabling advanced capabilities such as monitoring, simulation, and analytics. However, implementing DTs in large-scale, complex, and distributed environments like power systems introduces challenges, including latency, bandwidth limitations, and scalability issues. This paper proposes an edge-fog-cloud framework for hierarchical DTs. By decomposing complex systems into manageable components, a virtual replica of the physical system is developed and operated across the edge, fog, and cloud layers. Edge computing handles localized data processing, fog computing supports regional simulations and analytics, and cloud computing manages large-scale storage and resource-intensive operations. The results highlight notable improvements in latency and overall system efficiency, demonstrating the proposed framework's effectiveness as a scalable and reliable solution for managing complex, distributed DT systems.**

***Index Terms*—Digital Twins (DT), Edge-Fog-Cloud Framework, Hierarchical Digital Twins, Internet of Things (IoT).**

I. Introduction

Digital Twin (DT) technology has emerged as a transformative innovation, enabling real-time virtual models of physical systems to enhance monitoring, simulation, and predictive analytics [1]. By seamlessly bridging the physical and digital domains, DTs optimize operations, improve decision-making, and enhance system performance across industries such as energy, manufacturing, and transportation [2].

Power systems, as complex and expansive networks encompassing generation, transmission, and distribution, presents a compelling use case for DT technology [3]. Managing vast geographical regions and processing immense data volumes, it requires real-time monitoring, advanced simulations, effective control, and AI-driven capabilities to ensure stability, reliability, and efficiency. Integrating DTs into this critical infrastructure unlocks significant opportunities for optimizing energy generation, distribution, and consumption, improving renewable energy integration, and promoting sustainable operations [4].

However, applying DTs to large-scale distributed systems like power systems presents several challenges [5]. First, modeling such complex and dynamic systems requires robust methodologies to ensure reliability and seamless integration with real-world operations. Second, maintaining efficiency at scale demands optimized computational resources and intelligent data management strategies. Finally, real-time data collection remains a critical hurdle, as traditional cloud-based solutions often struggle with high latency, bandwidth limitations, and scalability constraints.

To address the complexities of simulating large-scale systems, hierarchical approaches have been proposed. Qi et al. [6] introduced a hierarchical DT approach in manufacturing, structured into asset-level, system-level, and system-of-systems (SoS) levels. The asset-level models individual components, the system-level integrates multiple asset-level DTs to simulate coordinated operations of individual systems, and the SoS-level connects several systems to enable global coordination and decision-making. Similarly, Tao et al. [7] utilized the same hierarchical DT structure. Asset-level DTs captured the geometric features, functional behavior, and operational status of individual components. The system-level combined these models to represent complete production lines or shop floors. At the SoS-level, multiple system-level DTs were integrated across various stages of the product life cycle to support data sharing, traceability, and collaboration between enterprises.

Simultaneously, to enhance DT systems in terms of computational efficiency and data collection, hybrid edge-fog-cloud frameworks have been developed. These frameworks aim to overcome the limitations of traditional cloud-centric models by distributing data processing across multiple layers to mitigate latency, bandwidth, and scalability issues. For instance, Knebel et al. [8] integrated fog nodes into DTs to minimize latency and reduce network congestion. Kalyani et al. [9] proposed a hybrid edge-fog-cloud framework for smart agriculture.

In our previous work, we developed a structured DT development model tailored for complex systems, providing a systematic approach for the planning, requirement analysis, design, implementation, operation, and maintenance of DTs within a unified and secure solution [10]. Each micro-grid component was individually modeled and integrated within a cloud environment to represent the physical micro-grid. Real-time data from the physical system was incorporated to ensure accurate synchronization between the physical and digital counterparts. The proposed reference DT architecture standardized the integration and operation of DTs and incorporated key DT services such as simulation, configuration, scenario analysis, and data analytics. The cloud management layer orchestrated several cloud providers through DT templates, and the DT management layer coordinated key DT operations such as simulation, configuration, and data analytics.

However, existing studies in the field of DTs often overlook the real-time data collection, responsiveness, and scalability requirements of DT systems for complex and geographically dispersed systems, and fail to offer DT solutions that effectively address these challenges. This paper extends our previous work [10], [11] and proposes a distributed framework for hierarchical DTs. In the hierarchical DT approach, the asset-level DT models individual components of the physical system, the system-level DT integrates multiple asset-level DTs to simulate coordinated system operations, and the SoS-level DT coordinates multiple system-level DTs to enable global system representation and management. This hierarchical DT structure is deployed within a distributed edge-fog-cloud framework to enhance efficiency and scalability. The edge layer handles real-time data collection and processing, the fog layer supports regional simulations and data analytics, and the cloud layer oversees global coordination, large-scale storage, and computationally intensive tasks. Experimental results demonstrate that this framework significantly reduces latency, improves system efficiency, and enhances scalability, providing a robust and efficient solution for complex physical systems.

The primary contributions of this work are as follows:

- A hierarchical, service-oriented DT approach is proposed to enable efficient simulation of the power system by decomposing its complex structure into manageable components, thereby enhancing system management and streamlining real-time data collection.
- A hybrid edge-fog-cloud framework is developed to support the efficient deployment and management of DT components in distributed environments, utilizing orchestration techniques for resource allocation across layers.
- A set of key architectural components is integrated to manage the operation of the DT system, ensuring efficient resource allocation, robust security, and maintaining scalability and reliability.
- The effectiveness of the proposed solution is validated through experiments, demonstrating significant improvements in latency, scalability, and efficiency for large-scale distributed systems.

The remainder of the paper is structured as follows: Section II reviews related work on power systems and DT concepts. Section III describes the replicated physical system, introduces the proposed hierarchical DT approach, and outlines the hybrid edge-fog-cloud framework along with its key components. Section IV presents the experimental results and their evaluation. Section V offers a detailed analysis of the findings, and Section VI summarizes the paper.

II. Background

A. Digital Twins

A DT is a virtual representation of a physical system, facilitating operational efficiency, fault detection, and decision-making. It has become a transformative technology in Industry 4.0, including manufacturing and smart grids, enabling real-time monitoring, simulation, predictive analytics, and optimization across various sectors. In the energy sector, DTs offer advanced capabilities for grid management, renewable energy integration, and system optimization as AI-driven, real-time adaptive systems that incorporate IoT devices, sensors, and cloud platforms [1].

Recent studies highlight the advantages and challenges of DTs in energy systems. Bazmohammadi et al. [12] survey DT applications in micro-grids, outlining key functionalities and related challenges. Kumari et al. [13] review DT use in grid-connected micro-grids, focusing on component-level applications and integration needs. Bassey et al. [14] explore DTs in renewable micro-grids, highlighting their benefits and implementation challenges. Jafari et al. [15] examine DT roles across smart grids and cities, emphasizing system intelligence and integration barriers. Cioara et al. [16] categorize DT applications in smart grids and point to the lack of unified frameworks. Mchirgui et al. [17] review DTs in smart grids with a focus on operational use cases and technical requirements. Sifat et al. [18] discuss DT functions in electric grids and related technical challenges. Das et al. [19] review DTs in the energy sector, emphasizing ML integration and system-level concerns.

Based on these studies, DTs demonstrate clear advantages across critical operational domains in energy systems. First, real-time monitoring is enabled, maintaining continuous synchronization with the physical system and allowing operators to track system status and respond immediately to changes. Second, DTs can perform predictive analysis by utilizing historical trends and live data to forecast future system conditions, supporting timely and informed planning. Third, DTs enable early anomaly detection, reducing downtime and preventing cascading failures. Lastly, DTs contribute to system optimization by analyzing performance in real time and supporting adaptive control strategies that improve efficiency, stability, and resource utilization.

Despite their advantages, DTs face several challenges that limit their widespread implementation. Ensuring data accuracy and consistency is a fundamental concern, particularly when integrating diverse sources and maintaining real-time synchronization between the physical and digital spaces. Interoperability remains a major challenge, as DTs must operate across heterogeneous devices, and communication protocols. Scalability further complicates deployment as system complexity increases, requiring more sophisticated data management strategies and computational resources. The lack of standardized frameworks and guidelines also hampers the consistent development, deployment, and management of DTs across distributed environments. Cyber-security poses a critical risk, as vulnerabilities in data transmission and system control can threaten operational stability and expose systems to external attacks. Additionally, the high costs associated with developing and maintaining DT infrastructures, including sensor networks, cloud services, and edge computing platforms, present significant barriers.

B. Hierarchical DT Approach

Physical systems comprise interconnected components operating at different scales, necessitating advanced modeling to capture their interactions effectively. Single-layer DT solutions often fall short due to component diversity, complex dependencies, and large volumes of real-time data. The proposed hierarchical DT approach overcomes these challenges by structuring digital representations across multiple levels, enabling individual modeling, data collection, analysis, and control while integrating them into a comprehensive model.

Several studies have utilized hierarchical DT approaches across various domains. In the manufacturing domain, Qi et al. [6] and Tao et al. [7] introduced a hierarchical DT model organized across asset, system, and SoS levels. Asset-level DTs monitored individual components, system-level DTs managed coordinated operations among asset-level DTs, and SoS-level DTs enabled global integration and cross-system coordination. Lippi et al. [20] presented a hierarchical model focused on manufacturing that spans from sensor-level data acquisition to plant-wide abstraction, improving transparency and enabling causal reasoning through layered analysis. Slot et al. [21] proposed a recursive, modular approach based on layered virtualization, allowing for nested DT composition and flexible configuration through digital states, simulations, and design masters. Shangguan et al. [22] proposed a hierarchical DT approach for CPS design that progresses from component-level DTs for individual assets to system-level simulation and application-level assessments, enabling dynamic model updates and reuse. Sado et al. [23] extended the hierarchical DT concept to naval power systems by structuring DT blocks that encapsulate individual subsystems at the component level, organize coordinated subsystems at the intermediate level, and manage overall system behavior at the SoS level. Ruhe et al. [24] proposed a hierarchical DT approach for energy systems, structuring digital representations across multiple levels to enhance system monitoring, simulation, and decision-making, thereby enabling a more organized and scalable approach to analyzing complex systems.

These works demonstrate that hierarchical DT approaches support structured modeling, enhance adaptability, and improve scalability through modular encapsulation and supervisory control. However, most implementations remain domain-specific and lack a general-purpose DT solution that integrates real-time monitoring, scenario analysis, physical system control, and data analytics. This work addresses this gap by proposing a hierarchical, service-oriented approach within a unified and secure DT solution.

C. Distributed Frameworks for Digital Twins

Energy systems span large geographical areas and integrate diverse resources, presenting challenges for data collection, computational efficiency, and scalability. Traditional cloud-based DT solutions often fall short in meeting real-time requirements due to centralized processing. Distributed solutions address these issues by distributing workloads across multiple processing nodes, which are strategically positioned close to the regions of interest. Performing simulations near data sources at localized and intermediate layers improves simulation efficiency, enhances data processing capabilities, and increases scalability.

Several studies have explored distributed DT frameworks to enhance performance and scalability. For instance, Knebel et al. [8] developed a micro-service based DT framework distributed between fog and cloud layers, achieving more than 50% latency reduction. Kalyani et al. [9] applied a similar fog-cloud approach to smart agriculture, demonstrating adaptive DT performance across field, sub-field, and farm levels. In manufacturing, Li et al. [25] and Lin et al. [26] proposed frameworks that allocate DT tasks across edge, fog, and cloud layers, enabling low-latency processing at the edge, analytics and coordination at the fog, and strategic planning at the cloud. Abdullahi et al. [5] presented a DT framework for predictive maintenance in wind farms, distributing asset-level DTs to the edge for local monitoring, system-level DTs to the fog for condition analysis, and SoS-level DTs to the cloud for global forecasting. Similarly, Yu et al. [27] developed an edge intelligence–driven DT framework for CNC systems, enabling real-time deployment and distributing tasks dynamically based on system requirements.

Despite the progress, a comprehensive DT solution for distributing DT components and services across an edge-fog-cloud framework remains lacking. Existing implementations are often domain-specific, lack standardized orchestration mechanisms, and do not fully integrate core DT functionalities such as simulation, configuration, scenario analysis, real-time data collection, physical system control, and data analytics within a unified and secure framework. This work addresses these limitations by proposing a modular, service-oriented DT approach that decomposes physical systems into asset, system, and SoS levels, and strategically distributes these workloads across edge, fog, and cloud layers. The following section provides a detailed explanation of the proposed DT approach.

III. Proposed DT Approach

The proposed approach establishes a modular virtual representation of the physical system by integrating hierarchical DT approach within a distributed edge-fog-cloud framework. Leveraging the hierarchical DT approach, the power system is decomposed into asset-level, system-level, and SoS-level DTs. These DT components are strategically deployed across the edge, fog, and cloud layers. The edge layer manages localized data collection and processing, the fog layer performs regional simulations and intermediate analytics. The cloud layer oversees large-scale storage and computationally intensive analytics. This layered framework facilitates efficient coordination of DT components in distributed environments, improves system responsiveness, and enhances scalability across complex and distributed systems. The following subsection provides a detailed explanation of the simulated physical system, the hierarchical DT approach, and the distributed DT framework.

A. Physical System

Figure 1 illustrates the modeled power system as a network of interconnected regional subsystems, each consisting of thermal power plants, renewable energy sources (RES), BSS, consumers, and control systems. Thermal power plants, including coal-fired, natural gas, and nuclear facilities, provide large-scale and stable electricity generation, while RES, such as solar and wind, contribute to sustainable energy production but introduce variability due to environmental factors. BSS play a crucial role in maintaining the stability by storing excess energy generated during periods of low demand or high RES output and releasing it when demand exceeds generation. Consumers across residential, commercial, and industrial sectors define electricity demand. Control systems monitor and manage grid components to ensure the efficient and stable operation of regional systems.

The grid operates under dynamic conditions shaped by internal factors such as fluctuations in generation and load, and external influences including weather conditions and market variability. Certain regions function primarily as producers, generating more electricity than they consume by leveraging local energy sources like wind and solar. In contrast, areas with dense residential and industrial activity act as major consumers with comparatively lower local generation capacity. Transmission lines and substations interconnect these regional subsystems, facilitating efficient electricity transfer while minimizing transmission and conversion losses.

Various IoT-based sensors, smart meters, and monitoring systems enable real-time data collection from the physical system. These sensors, deployed across both the production and consumer sides, as well as control systems, continuously monitor electricity generation and consumption. Additionally, substations and transmission lines provide both corrective and additional measurements. External factors such as weather systems and energy markets also provide data. These data streams are transmitted over communication networks using standardized communication protocols.

Despite significant technological progress, data collection, processing, and storage continue to pose challenges. The wide range of heterogeneous data sources, coupled with inconsistent sampling rates, makes integration complex. Efficient processing methods are required to handle this complexity while ensuring seamless interoperability. Additionally, the increasing volume of data necessitates scalable storage solutions that support real-time accessibility and long-term historical data preservation. Privacy and security concerns further add to these challenges, requiring robust protection mechanisms.

In this research, a hierarchical DT approach is used to replicate the physical power system. The physical system is decomposed into multiple layers utilizing a structured and hierarchical layout. Operational and external data are integrated into the DTs to ensure accurate representation and synchronization. The following subsection details the hierarchical DT approach.

B. Hierarchical DT Approach

This research introduces a hierarchical DT approach for modeling complex systems, addressing challenges in simulation and data management through the structured decomposition of components. The power system is modeled across asset, system, and SoS levels. Asset-level DTs represent individual physical components, system-level DTs integrate and manage multiple asset-level DTs, and the SoS-level DT offers a comprehensive perspective by coordinating system-level DTs. This structured hierarchy supports a dynamic and holistic representation of the physical power system, while promoting efficient management, modularity, seamless integration, and scalability. Figure 2 illustrates the hierarchical structure of DT components and their interactions. The following subsections provide a detailed explanation of each hierarchical level.

1) Asset-level DTs: Asset-level DTs simulate individual system components, including thermal power plants, RES, BSS, consumers, controllers, weather systems, and the energy market. Developing each asset-level DT involves analyzing the corresponding physical component, considering its parameters, operational behavior, and interactions within the system. These DTs function independently while delivering essential services that support the broader DT system. Their primary responsibilities include simulation, configuration, scenario analyses and data analytics. In addition, they enable data collection and processing, as well as control of the physical system. Equation 1 represents the net power of each asset-level DT by capturing the difference between power generated P_{gen} and power consumed P_{cons}.

$$DT_{asset-level,n} = P_{gen,n} - P_{cons,n} \tag{1}$$

2) System-level DTs: System-level DTs simulate regional power system operations by integrating and managing multiple asset-level DTs. The interactions among asset-level DTs are modeled to replicate the dynamic behavior of a regional power system. Data from asset-level DTs are utilized to perform localized system simulations, analyze overall system behavior, and generate analytical outputs. Similar to asset-level DTs, system-level DTs provide services for simulation, configuration, scenario analyses, physical system control, and data analytics. Equation 2 models the system-level DT as the summation of the net outputs of all associated asset-level DTs, capturing the aggregated operational behavior of the regional power system.

$$DT_{system-level,m} = \sum_{n=1}^{N} DT_{asset-level,n} \tag{2}$$

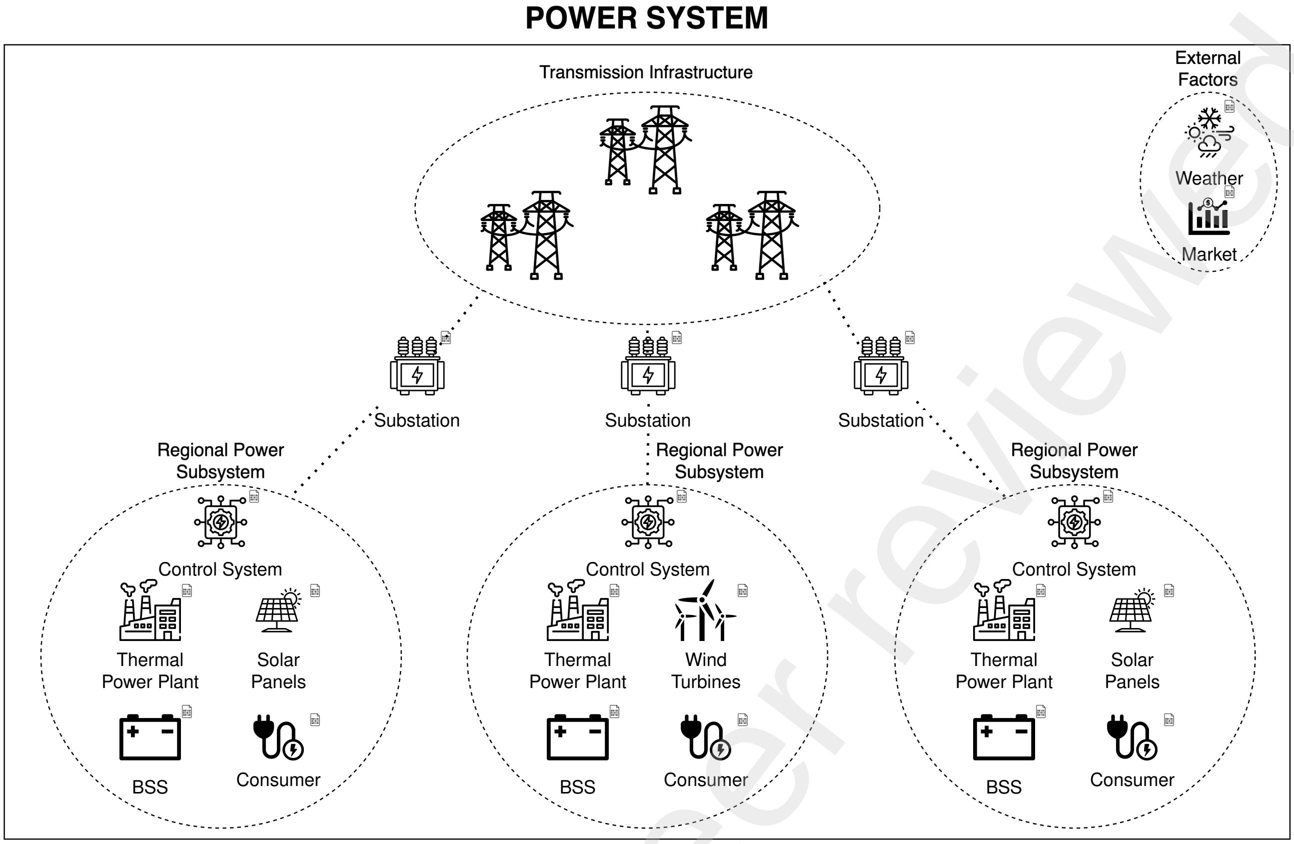

Fig. 1: Physical Power System

3) SoS-level DTs: At the SoS level, a holistic view of the power system is provided by managing multiple system-level DTs, with a focus on large-scale integration and analysis. By coordinating multiple system-level DTs, the simulation of the power system is created. Similar to asset-level and system-level DTs, SoS-level DTs provide services for simulation, configuration, scenario analyses, physical system control, and complex data analysis tasks. Equation 3 formulates the aggregated impact of system-level DTs for global coordination, representing the overall behavior of the entire power system.

$$DT_{sos-level} = \sum_{m=1}^{M} DT_{system-level,m} \quad (3)$$

This hierarchical organization of DTs enables a structured approach to managing the complexities of the physical system. Each level addresses specific objectives, from individual component simulations at the asset-level, to regional coordination and operational analysis at the system-level, and comprehensive system-wide coordination and strategic analysis at the SoS-level. Simulations, configuration, scenario analysis, real-time data collection and processing, and data analytics are supported at each level. Additionally, efficiency and scalability are enhanced through the strategic deployment of DT components across the edge, fog, and cloud layers. The next subsection provides further details on the edge-fog-cloud framework.

C. Edge-Fog-Cloud Framework

The developed hierarchical DT approach creates modular DT components that collectively enable the systematic simulation of the power system. However, executing all DT components within a single computational environment is inefficient and limits scalability, responsiveness, and real-time processing capabilities. To overcome this limitation, this study proposes a distributed DT framework that deploys DT components across the edge, fog, and cloud layers.

Figure 3 illustrates the proposed distributed DT framework, comprising the edge, fog, and cloud layers. This multi-layered structure improves computational efficiency, supports scalability, and enhances real-time processing capabilities, enabling an optimized and responsive simulation of the power system. The following subsections provide a detailed description of each layer within the proposed edge-fog-cloud framework.

1) Edge Layer: The edge layer serves as the primary interface between the physical system and the DT. Edge devices, strategically positioned near key grid components, collect real-time data from physical system components and transmit it to asset-level DTs deployed in the fog layer. In addition to data collection, edge devices perform essential preprocessing tasks, including filtering, aggregation, and cleansing of raw data. Additionally, edge devices play a crucial role in the control feedback loop, enabling two-way communication between the physical system and the DT.

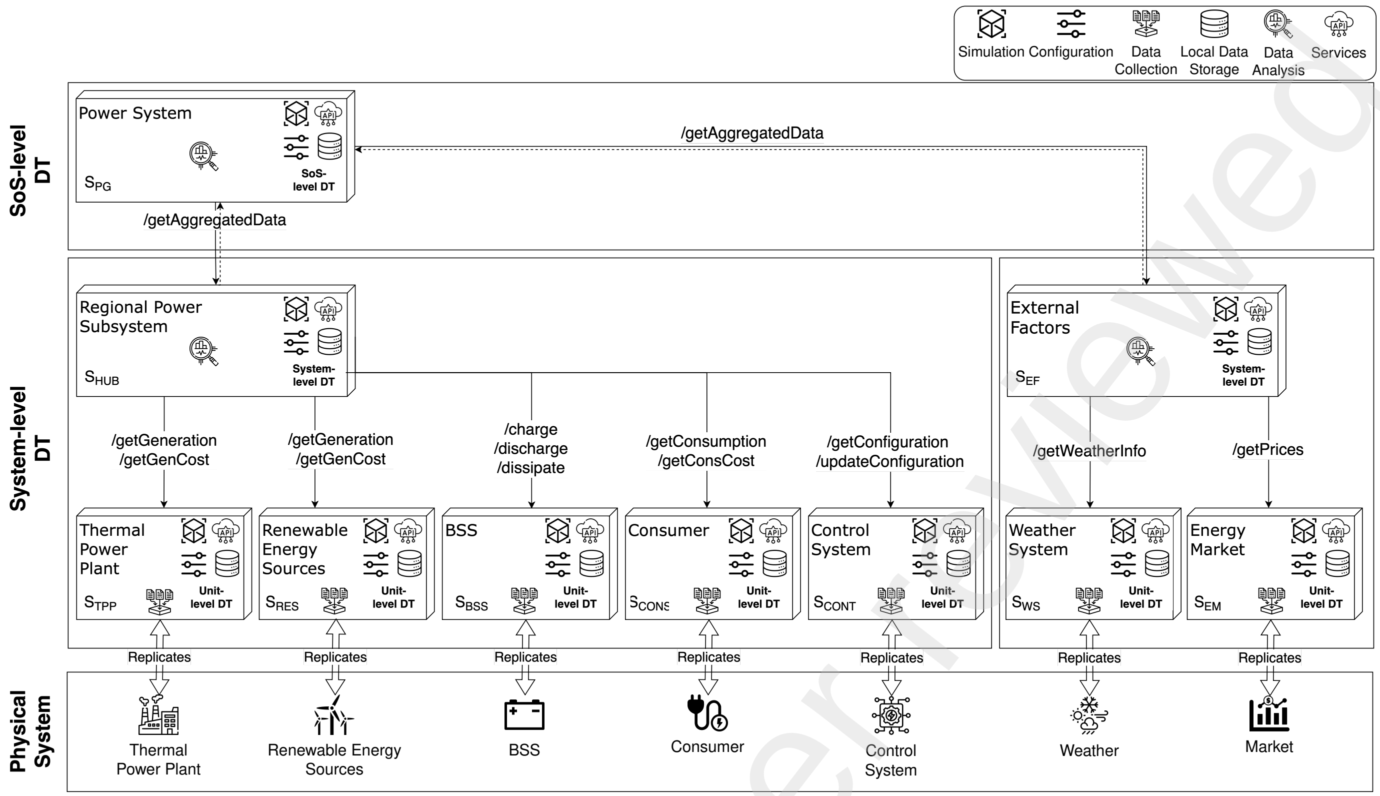

Fig. 2: Hierarchical Structure of DT Components and Their Interactions

2) Fog Layer: The fog layer serves as a bridge between the edge and cloud layers. Positioned closer to the physical systems than the cloud, it reduces bandwidth usage and latency while enhancing system responsiveness. In the proposed framework, distributed simulations are run within the fog layer to create accurate and scalable virtual replicas of regional power systems. Data transmitted from edge devices is stored by asset-level DTs and later retrieved by system-level DTs to generate system-level simulations and analyses.

3) Cloud Layer: The cloud layer acts as the central hub in the edge-fog-cloud framework, managing global DT operations and handling resource-intensive tasks, including large-scale data storage and advanced data analysis. It ensures reliable data redundancy and long-term data preservation. By running the core DT services proposed in our previous work [10], it manages DT resources, integrates key services, and provides user interfaces. It coordinates global DT operations, including simulation, configuration, scenario analyses, physical system control, and complex data analytics. Robust user interfaces facilitate visual interaction with the system and enable high-level control functionalities.

The proposed method enables digital representation through two key processes, data collection and simulation. Data collection follows a bottom-up approach, where edge devices gather real-time data from physical components and transmit it to asset-level DTs. The simulation process follows a top-down approach, beginning at the SoS-level DT and cascading down to system-level and asset-level DTs.

Equation 4 defines the total time required for edge devices to collect and transmit data. This process involves two main delays, $T_{\text{edge-physical}}$, which represents the time needed to acquire data from physical components, and $T_{\text{edge-fog}}$, which accounts for the time required to transfer the data to the fog layer.

$$T_{\text{data collection}} = T_{\text{edge-physical}} + T_{\text{edge-fog}} \tag{4}$$

Once the SoS-level DT initiates the simulation, system-level and asset-level DTs execute their local simulations based on the collected data. During this process, both computational and transmission delays occur. The total simulation latency, considering these factors, is defined by Equations 5 and 6.

$$S_{\text{system-level}} = \sum_{n=1}^{N} \left(S_{\text{asset-level},n} + T_{\text{fog-fog},n} \right) \tag{5}$$

$$S_{\text{SoS-level}} = \sum_{m=1}^{M} \left(S_{\text{system-level},m} + T_{\text{cloud-fog},m} \right) \tag{6}$$

In these equations, $S_{\text{asset-level}}$, $S_{\text{system-level}}$, and $S_{\text{SoS-level}}$ denote the computational delays at each level, while $T_{\text{cloud-fog}}$ and $T_{\text{fog-fog}}$ account for the data transmission latencies between DTs. The parameters N and M represent the number of asset-level DTs and system-level DTs, respectively.

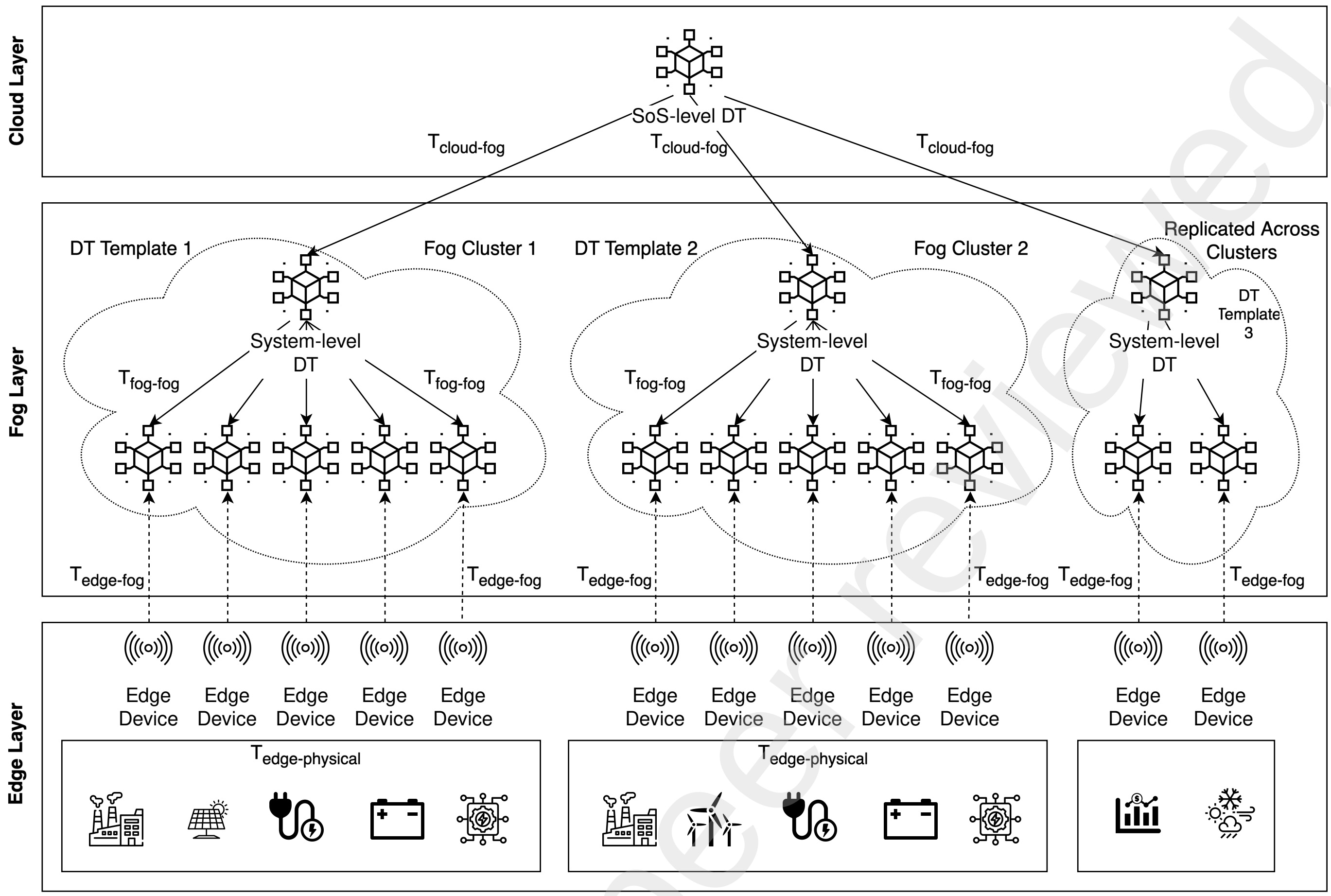

Fig. 3: Distributed Edge-Fog-Cloud Framework for Hierarchical DTs

D. Key Components of the Distributed DT Framework

The distributed DT framework is managed by several key components that collaborate to ensure seamless operation, scalability, and security. These components are outlined below.

1) Kubernetes: The proposed framework utilizes Kubernetes [28] as the virtualization platform for both the cloud and fog layers. Kubernetes manages DT workloads at the asset, system, and SoS levels, enabling controlled deployment and scaling while ensuring high availability and resource efficiency.

2) Repository: The proposed framework utilizes a central DT repository as a storage solution for managing and tracking DT images. By allowing users to store and share DT images in a repository, it enables efficient collaboration. Access to the repository is controlled based on roles and permissions, ensuring secure and organized management of DT images.

3) API Gateways: The proposed framework utilizes API Gateways to manage communication between the physical system and the DT system. API Gateways are deployed within each cluster in the fog and cloud layers, where data from the physical system is routed to the appropriate DT instances. Authentication and authorization mechanisms are enabled to secure communications, ensuring that only verified requests access the DT system. Additionally, load balancing and efficient scaling of DT instances within each layer are supported.

4) Authentication and Authorization: The proposed framework utilizes distributed authentication and authorization mechanisms to secure interactions between the DT system, users, and the physical system. Advanced authentication is implemented at each API Gateway to ensure that the DT system is properly secured.

5) System Monitoring: The proposed framework utilizes monitoring mechanisms to track compute, storage, and network resources, as well as system operations. Several metrics such as CPU usage, memory usage, and disk reads/writes are continuously collected and analyzed. This enables users and administrators to understand and optimize system resources and performance. Additionally, all system operation activities are logged to support auditing, troubleshooting, and system performance evaluation.

6) Security: The proposed framework utilizes comprehensive security mechanisms to protect the DT system. Security measures are implemented at multiple layers and interfaces to safeguard data, services, and infrastructure from both external and internal threats. Data in transit is encrypted, regular security audits are conducted, and system monitoring is employed to defend against potential cyber threats.

IV. EXPERIMENTAL RESULT AND EVALUATION

The effectiveness and applicability of the proposed DT framework were validated through detailed experiments conducted in a controlled simulation environment. The experiments aimed to assess its performance in real-time data collection, processing efficiency, and scalability. The evaluation focused on the framework's ability to handle real-time data streams and to scale with an increasing number of DT instances.

A. Experimental Setup

The physical power system was modeled through the proposed hierarchical DT approach at the asset, system, and SoS levels. Asset-level DTs were created for each physical component, including thermal power plants, RES, BSS, consumers, and control systems. System-level DTs represented regional power systems by managing multiple asset-level DTs. The SoS-level coordinated multiple system-level DTs, each corresponding to distinct energy regions. Additionally, external entities such as weather systems and the energy market were encapsulated within a dedicated system-level DT.

Each DT at each hierarchical level was encapsulated within a separate Docker image that provided essential DT services and was deployed to the compute clusters. Next, DT template configuration files were created for system-level DTs representing regional power systems and external systems. These templates defined deployment settings and network layouts. Finally, the configuration files were deployed to the clusters in the fog and cloud layers. Once the DT instances became fully operational, the DT system was capable of receiving real-time data, performing localized simulations, and generating operational insights.

TABLE I: Key Parameters within the DT Framework

Parameter	Initial Value	Description
Cluster Configuration		
N_{cluster}	3	Number of fog clusters.
N_{nodes}	3	Number of nodes in each fog cluster.
D_{type}	Replicated	Deployment type (Single or Replicated).
Communication Latencies		
$T_{\text{cloud-fog}}$	$\sim$100 ms	Latency between cloud and fog layers.
$T_{\text{fog-fog}}$	$\sim$10 ms	Latency within the fog layer.
$T_{\text{edge-fog}}$	$\sim$20 ms	Latency between edge devices and the fog layer.
$T_{\text{edge-physical}}$	$\sim$5 ms	Latency between edge devices and physical system components.
Data Sizes		
D_{TPP}, D_{RER}, D_{CONS}, D_{PRO}, D_{CONT}	<1 KB	Transferred data sizes.

The experiment was designed to evaluate the performance of the edge-fog-cloud framework under various conditions. Table I presents key parameters within the distributed framework, focusing on cluster configuration, communication latencies, and data sizes. Cluster configuration defines the structural organization of the clusters, including the number of fog clusters, the number of nodes within each cluster, and the deployment type. Communication latencies correspond to the average delays measured in the current deployment. Data sizes represent the amount of information exchanged between edge devices and asset-level DTs. These parameters were carefully analyzed to assess the behavior of the DT framework under varying conditions and to evaluate its responsiveness, scalability, and overall performance.

B. Dataset

The experiments utilize real data from the Turkish Power Grid, covering multiple energy regions and incorporating various energy sources, including thermal power plants and RES. The energy data is sourced from the YTBS [29], while the energy market data comes from the Energy Exchange Transparency Platform [30]. Each dataset begins on January 1, 2025, and provides a continuous record of key grid operations, capturing power generation, consumption, and total energy exchange. Additionally, meteorological conditions and market prices are included, as they significantly influence grid performance.

TABLE II: Statistical Summary of Power System Data

Metric	Mean	Min	Max	Std Dev
Region 1				
Generation (MW)	178.30	0.00	299.00	113.83
Consumption (MW)	92.87	8.00	181.67	52.35
Temperature (°C)	4.11	-6.60	14.40	3.55
Region 2				
Generation (MW)	2.62	0.00	17.96	4.68
Consumption (MW)	26.93	-35.04	82.40	24.46
Temperature (°C)	3.62	-6.90	14.90	4.39
Region 3				
Generation (MW)	30.17	-1.37	114.13	28.95
Consumption (MW)	50.93	19.67	97.60	17.65
Temperature (°C)	8.27	-1.30	21.40	4.28
Market Price				
Price ($/MWh)	68.40	15.25	82.97	11.85

Table II presents statistical insights. In Region 1, power generation averages 178.30 MW, peaking at 299 MW, while consumption averages 92.87 MW, with a maximum of 181.67 MW. Region 2 has lower generation, averaging 2.62 MW, while consumption fluctuates between -35.04 MW and 82.40 MW, reflecting variations in energy trade. Region 3 generates an average of 30.17 MW, with a peak of 114.13 MW, while consumption ranges from 19.67 MW to 97.60 MW. Environmental conditions also vary, with average temperatures of 4.11°C in Region 1, 3.62°C in Region 2, and 8.27°C in Region 3.

C. Latency Analysis

The latency analysis assessed the framework's capability for real-time data collection and processing. Critical data were collected from multiple regions, each consisting of dedicated edge devices. The edge devices transmitted data gathered from physical components to asset-level DTs operating in the fog layer. The tests focused on the impact of the number of nodes in each fog cluster, different deployment types (whether each container was deployed to a single node or across all nodes in the cluster), varying distances between the edge and fog layers, and different data sizes for transmission. In each test, the number of data transmissions gradually increased to evaluate the system's reliability under growing loads.

TABLE III: Quantitative Latency Analysis Across Different Cluster Configurations

Configuration	Mean (s)	Median (s)	Min (s)	Max (s)	Std Dev (s)
Number of Nodes					
1 Node	9.913	10.319	0.229	20.847	5.595
2 Node$^{(\text{Single})}$	6.362	6.236	0.090	17.820	3.991
3 Node$^{(\text{Single})}$	4.284	3.837	0.099	14.601	2.939
Number of Nodes (Replicated)					
2 Node$^{(\text{Replicated})}$	5.659	5.446	0.095	32.168	3.598
3 Node$^{(\text{Replicated})}$	**3.604**	**3.588**	**0.052**	**9.798**	**2.108**
Network Latency					
100 ms	5.767	5.574	0.846	13.572	3.343
200 ms	8.293	8.137	1.641	23.764	4.632
400 ms	12.670	13.278	3.225	31.992	6.391
Data Size					
1 KB	4.761	4.699	0.125	16.006	2.926
10 KB	8.085	8.100	0.177	18.511	4.734
100 KB	13.169	12.920	0.269	28.917	7.675

Figure 4 presents the average latency results, while Table III provides a statistical summary, including mean, median, minimum, maximum, and standard deviation. The results highlight that cluster configuration, deployment strategy, network conditions, and data size significantly influence system performance.

The first test demonstrates that increasing the number of fog nodes in a cluster significantly reduces latency. A single-node configuration results in the highest average latency at 9.913 seconds, decreasing to 6.362 seconds with two nodes and further dropping to 4.284 seconds with three nodes.

The second test shows that deployment strategy also plays a crucial role. The replicated deployment model, where multiple instances of each DT are distributed across all fog nodes, reduces latency. In a two-node configuration, latency drops from 6.362 seconds in the single deployment model to 5.659 seconds in the replicated model. In a three-node setup, latency decreases from 4.284 seconds to 3.604 seconds, with lower standard deviation, indicating more consistent performance.

The third test examines the impact of network latency between edge and fog layers. As network delay increases, system latency rises significantly. With a 20 ms network delay, the average latency is 4.284 seconds. This increases to 5.767 seconds at 100 ms, 8.293 seconds at 200 ms, and 12.670 seconds at 400 ms.

Data size also affects performance, as larger transmissions lead to higher latency. When data size grows from 1 KB to 10 KB, latency rises from 4.761 seconds to 8.085 seconds. At 100 KB, it reaches 13.169 seconds, reflecting the impact of transmission overhead.

Overall, the results indicate that increasing fog nodes and using a replicated deployment model can significantly reduce latency, though at the cost of higher resource consumption. Additionally, network latency and data size remain critical factors, emphasizing the need for efficient network and data management strategies to ensure reliable real-time system performance.

D. Scalability Analysis

Scalability analysis evaluated the framework's ability to manage an increasing number of SoS-level DTs with dedicated system-level and asset-level DTs. Deployments with 1, 2, and 3 SoS-level DTs were tested, with CPU and memory usage monitored at each stage. The number of fog nodes within each cluster was increased to assess the system's scalability and its capacity to support future expansion.

The results in Figure 5 show that increasing the number of SoS-level DTs leads to higher overall resource usage, causing latency issues in the DT system. However, adding more fog nodes distributes the workload more efficiently, reducing resource usage per node and improving system responsiveness. In a single-node setup with three SoS-level DT deployments, CPU usage peaked at 78.72%, and memory usage reached 87.64%. Expanding to two nodes reduced CPU usage to 37.85% and memory usage to 51.69%. With three nodes, CPU usage dropped further to 24.73%, and memory usage to 40.03%. These results demonstrate that distributing workloads across multiple nodes improves scalability, resource efficiency, and system performance.

V. Discussion

The proposed solution introduces a novel approach by presenting an edge-fog-cloud framework for DTs. The complex physical system is decomposed into smaller, service-oriented components, with DTs developed and operated in a distributed manner across the edge, fog, and cloud layers. The integrated DT system supports simulation, configuration, scenario analysis, real-time data collection, and advanced data analytics. This distributed and modular design enables efficient management, scalability, and resource-efficient virtual replication of physical systems.

The hierarchical DT approach specifically decomposes the complex physical system into structured layers at the asset, system, and SoS levels. Asset-level DTs simulate individual components, system-level DTs coordinate regional operations by integrating multiple asset-level DTs, and SoS-level DTs manage system-wide coordination across regions. Additionally, the modular microservice-based design enables flexible development, integration, and deployment, allowing each DT component to be developed, tested, and maintained independently while supporting continuous system improvement through independent updates and refinements.

The DT system is strengthened by providing a comprehensive suite of services at each hierarchical layer. Services such as result retrieval, simulation, real-time monitoring, physical system control, advanced scenario analysis, and AI-driven

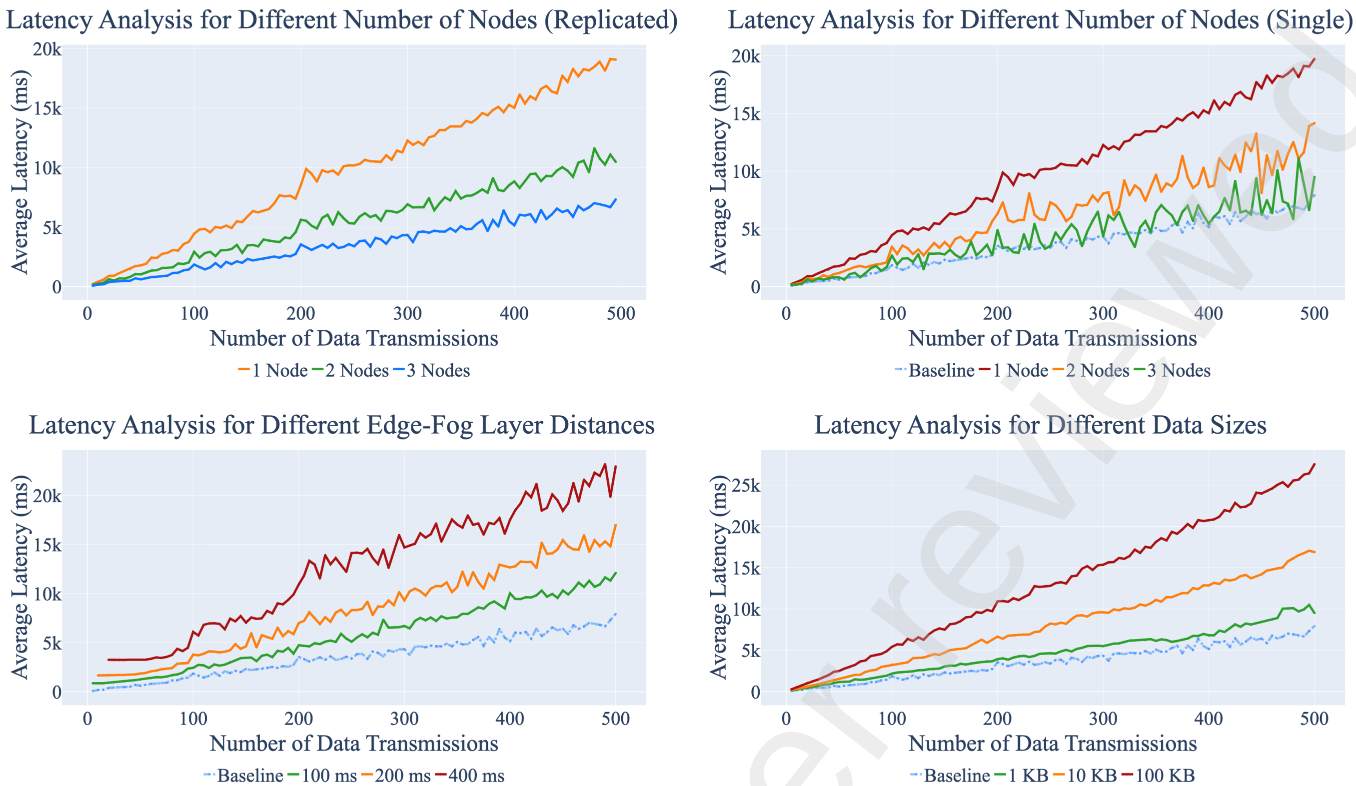

Fig. 4: Latency Analysis for Different Cluster Configurations

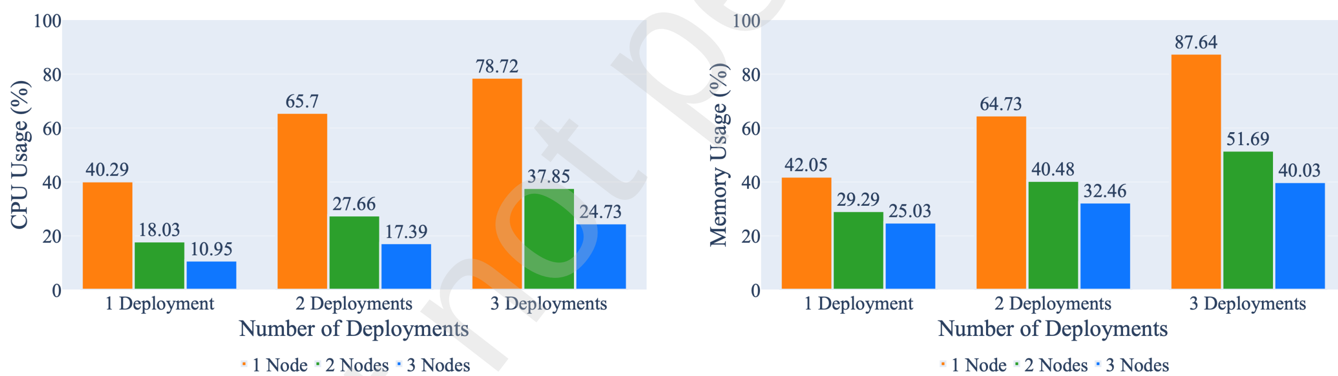

Fig. 5: CPU and Memory Utilization Across Different DT Deployments for Various Cluster Configurations

insights are available at each level, enabling extensive customizability for the developed DTs.

The edge-fog-cloud framework plays a critical role in enhancing real-time data collection and processing. Edge-layer devices collect real-time data from the physical system, the fog layer handles regional simulations and localized data analytics, and the cloud layer manages large-scale storage and computationally intensive tasks. By strategically positioning computational resources closer to data sources, the framework reduces bandwidth consumption and improves system responsiveness.

The use of DT template configuration files simplifies the deployment of DTs across distributed layers, while also ensuring consistency across DT deployments, thereby reinforcing system security and operational integrity. The integrated services for managing the edge, fog, and cloud layers streamline the efficient handling of distributed DT workloads.

The proposed framework is supported by several core components that ensure scalability, security, and operational efficiency. Kubernetes manages the deployment and scaling of DT workloads across the fog and cloud layers. A centralized DT repository provides consistent storage and version control of DT images. API Gateways secure communication between the physical system and DT instances. Authentication and authorization mechanisms protect system interactions. Integrated system monitoring tracks performance and resources, while multi-layered security measures safeguard the infrastructure against cyber threats.

Experiments on the proposed framework demonstrated its effectiveness in real-time data collection, processing, and scalability. The first set of experiments identified the optimal system configurations for real-time data collection with minimal latency, ensuring precise monitoring of physical components. The second experiment validated the framework's scalability, demonstrating its robustness in handling large-scale DT workloads. Overall, the proposed distributed edge-fog-cloud framework for hierarchical DTs enabled efficient power system simulations, ensuring effective resource management, data collection, and processing while enhancing scalability.

Despite its advantages, the proposed DT solution faces several challenges. One key challenge is the diversity of physical system components. Given the heterogeneous nature of these components, standardization and interoperability are essential for achieving accurate and scalable simulations. Additionally, the deployment of fog nodes in large-scale systems requires careful and optimal placement of computing nodes to maximize efficiency and minimize communication delays. Another significant challenge is maintaining real-time synchronization between the physical system and the DT under adverse network conditions.

Future work will focus on expanding the functionalities of the distributed hierarchical DT solution to support more advanced simulations and real-time control strategies. Integrating AI-driven analytics for predictive maintenance, fault diagnosis, and anomaly detection will further enhance decision-making capabilities. Additionally, research will explore adaptive workload distribution mechanisms to optimize the balance between distributed computing resources. Extending the DT framework to other domains, such as smart cities and industrial automation, will also be a key direction for future development.

VI. Conclusion

This study presents an edge-fog-cloud framework for hierarchical DTs to efficiently monitor, simulate, and manage complex physical systems. The hierarchical DT approach decomposes the physical system into structured components at the asset, system, and SoS levels, enabling modular and scalable digital replication. Asset-level DTs model individual physical components, capturing their operational behavior and real-time data. System-level DTs integrate multiple asset-level DTs to simulate regional system operations. SoS-level DTs manage and synchronize multiple system-level DTs, enabling system-wide simulation, monitoring, and comprehensive analysis across the entire physical system. The developed DTs are distributed across the edge-fog-cloud framework: the edge layer handles localized data collection and processing, the fog layer manages regional simulations and analytics, and the cloud layer supports large-scale storage and computationally intensive tasks. Experimental validation confirmed the system's effectiveness in real-time data collection, processing, and scalability, demonstrating its ability to support large DT workloads with minimal latency. The overall solution enhances power system management and contributes to DT research by advancing the development of real-time, data-driven DT infrastructures.

References

[1] A. Fuller, Z. Fan, C. Day, and C. Barlow, "Digital twin: Enabling technologies, challenges and open research," *IEEE Access*, vol. 8, pp. 108 952–108 971, 2020.

[2] M. R. Abayadeera and G. Ganegoda, "Digital twin technology: A comprehensive review," *International Journal of Scientific Research and Engineering Trends*, vol. 10, pp. 1485–1504, 07 2024.

[3] M. A. Yassin, A. Shrestha, and S. Rabie, "Digital twin in power system research and development: Principle, scope, and challenges," *Energy Reviews*, vol. 2, no. 3, p. 100039, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2772970223000263

[4] M. Singh, E. Fuenmayor, E. P. Hinchy, Y. Qiao, N. Murray, and D. Devine, "Digital twin: Origin to future," *Applied System Innovation*, vol. 4, no. 2, 2021. [Online]. Available: https://www.mdpi.com/2571-5577/4/2/36

[5] A. Abdullahi *et al.*, "Towards a distributed digital twin framework for predictive maintenance in industrial internet of things (iiot)," *Sensors*, vol. 24, no. 5, p. 1234, 2024. [Online]. Available: https://doi.org/10.3390/s24051234

[6] Q. Qi, D. Zhao, F. Tao, and T. W. Liao, "Modeling of cyber-physical systems and digital twin based on edge computing, fog computing and cloud computing towards smart manufacturing," in *ASME 2018 International Manufacturing Science and Engineering Conference*, 2018.

[7] F. Tao, Q. Qi, L. Wang, and A. Nee, "Digital twins and cyber–physical systems toward smart manufacturing and industry 4.0: Correlation and comparison," *Engineering*, vol. 5, no. 4, pp. 653–661, 2019.

[8] F. P. Knebel, J. A. Wickboldt, and E. P. de Freitas, "A cloud-fog computing architecture for real-time digital twins," *arXiv preprint arXiv:2012.06118*, 2020. [Online]. Available: https://arxiv.org/abs/2012.06118

[9] Y. Kalyani and R. W. Collier, "Digital twin deployment for smart agriculture in cloud-fog-edge infrastructure," *International Journal of Computer Integrated Manufacturing*, vol. 36, no. 8, pp. 825–841, 2023. [Online]. Available: https://doi.org/10.1080/17445760.2023.2235653

[10] E. Özkan, İbrahim Kök, and S. Özdemir, "System development lifecycle assisted digital twin development model for smart micro-grids," *Internet of Things*, vol. 31, p. 101580, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2542660525000939

[11] E. Özkan, Kök, and S. Özdemır, "Deeptwin: A deep reinforcement learning supported digital twin model for micro-grids," *IEEE Access*, vol. 12, pp. 196 432–196 441, 2024.

[12] N. Bazmohammadi, A. Madary, J. C. Vasquez, H. Bazmohammadi, B. Khan, Y. Wu, and J. M. Guerrero, "Microgrid digital twins: Concepts, applications, and future trends," *IEEE Access*, vol. 10, pp. 2284–2302, 2022.

[13] N. Kumari, A. Sharma, B. Tran, N. Chilamkurti, and D. Alahakoon, "A comprehensive review of digital twin technology for grid-connected microgrid systems: State of the art, potential and challenges faced," *Energies*, vol. 16, no. 14, p. 5525, 2023.

[14] K. E. Bassey, J. Opoku-Boateng, B. O. Antwi, A. Ntiakoh, and A. R. Juliet, "Digital twin technology for renewable energy microgrids," *Engineering Science & Technology Journal*, vol. 5, no. 7, pp. 2248–2272, 2024. [Online]. Available: https://www.fepbl.com/index.php/estj/article/view/1319

[15] M. Jafari, A. Kavousi-Fard, T. Chen, and M. Karimi, "A review on digital twin technology in smart grid, transportation system and smart city: Challenges and future," *IEEE Access*, vol. 11, pp. 17 471–17 489, 2023.

[16] T. Cioara, I. Anghel, M. Antal, I. Salomie, C. Antal, and A. G. Ioan, "An overview of digital twins application domains in smart energy grid," *arXiv preprint arXiv:2104.07904*, 2021. [Online]. Available: https://arxiv.org/abs/2104.07904

[17] N. Mchirgui, N. Quadar, H. Kraiem, and A. Lakhssassi, "The applications and challenges of digital twin technology in smart grids: A comprehensive review," *Applied Sciences*, vol. 14, no. 23, p. 10933, 2024. [Online]. Available: https://www.mdpi.com/2076-3417/14/23/10933

[18] M. M. H. Sifat *et al.*, "Towards electric digital twin grid: Technology and framework review," *Energy AI*, vol. 10, p. 100213, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2666546822000593
[19] O. Das, M. H. Zafar, F. Sanfilippo, S. Rudra, and M. L. Kolhe, "Advancements in digital twin technology and machine learning for energy systems: A comprehensive review of applications in smart grids, renewable energy, and electric vehicle optimisation," *Energy Conversion and Management: X*, vol. 24, p. 100715, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2590174524001934
[20] M. Lippi, M. Martinelli, M. Picone, and F. Zambonelli, "Enabling causality learning in smart factories with hierarchical digital twins," *Computers in Industry*, vol. 148, p. 103892, 2023.
[21] M. Slot, P. Huisman, and E. Lutters, "A structured approach for the instantiation of digital twins." *Procedia CIRP*, vol. 91, pp. 540–545, 2020, enhancing design through the 4th Industrial Revolution Thinking. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2212827120308635
[22] D. Shangguan, L. Chen, and J. Ding, "A hierarchical digital twin model framework for dynamic cyber-physical system design," in *Proceedings of the 2019 ACM/IEEE 10th International Conference on Cyber-Physical Systems (ICCPS)*, 2019.
[23] K. Sado, J. Hannum, E. Skinner, H. L. Ginn, and K. Booth, "Hierarchical digital twin of a naval power system," in *2023 IEEE Energy Conversion Congress and Exposition (ECCE)*, 2023, pp. 1514–1521.
[24] S. Ruhe, K. Schäfer, S. Branz, S. Nicolai, P. Bretschneider, and D. Westermann, "Design and implementation of a hierarchical digital twin for power systems using real-time simulation," *Electronics*, vol. 12, p. 2747, 06 2023.
[25] Y. Li *et al.*, "A reference framework for the digital twin smart factory based on cloud-fog-edge computing collaboration," *Journal of Intelligent Manufacturing*, 2024, details to be updated upon publication.
[26] Y. Lin *et al.*, "Digital thread driven cloud-fog-edge collaboration to mitigate production disturbances," *Journal of Manufacturing Systems*, 2024, details to be updated upon publication.
[27] T. Yu *et al.*, "Edge intelligence-driven digital twin of cnc system: Architecture and deployment," *Robotics and Computer-Integrated Manufacturing*, vol. 79, p. 102418, 2023. [Online]. Available: https://doi.org/10.1016/j.rcim.2022.102418
[28] C. N. C. Foundation, "Kubernetes," https://kubernetes.io/, 2025, accessed: 2025-02-18.
[29] YTBS, "Load dispatch information system," 2025, accessed: March 23, 2025. [Online]. Available: https://ytbs.teias.gov.tr/
[30] EPIAS, "Energy exchange transparency platform," 2025, accessed: March 23, 2025. [Online]. Available: https://seffaflik.epias.com.tr/