An Architectural Framework for 6G Network Digital Twin

Z. Yang, C. Papagianni, A.S.Z. Belloum, P. Grosso
Informatics Institute
University of Amsterdam
Amsterdam, The Netherlands
{z.yang,c.papagianni,a.s.z.belloum,p.grosso}@uva.nl

Abstract

In the context of 6G, Digital Twin technology has potential to play a crucial role by offering a sophisticated, dynamic model that mirrors the physical world in real-time. Integrating digital twins into 6G enables precise orchestration and optimization of network resources, meeting the diverse and stringent demands of next-generation applications. Furthermore, digital twins provide a platform for continuous learning and AI-driven insights. Research on the architecture of Network Digital Twins (NDTs) for 5G/6G is still limited, often focusing on partial implementations rather than comprehensive, full-stack approaches. This paper proposes a high-level architectural framework for 6G NDTs, structured across three layers: the Physical Twin Layer, Digital Twin Layer, and Application and Service Layer. We discuss the challenges of deploying these systems and the importance of selecting appropriate tools, presenting an experimental case study that demonstrates the impact of different tool choices on system performance. Our findings underscore the need for flexible, scalable solutions to fully realize the benefits of NDTs in 6G networks.

CCS Concepts

• **Networks** → **Network architectures**; **Network components**; **Network management**.

Keywords

6G, Mobile Network, Network Digital Twin, AI-driven Network

1 Introduction

The concept of Digital Twins (DTs) has seen significant growth in recent years, evolving from its initial introduction at a product lifecycle management conference to becoming a critical technology across various industries, particularly in manufacturing and aerospace. Despite its widespread adoption in these areas, the application and evaluation of DTs within network systems remain relatively underdeveloped. As the telecommunications industry gears up for 6G networks—offering unprecedented improvements in speed, capacity, and latency—the relevance of Network Digital Twins (NDTs) comes into focus [2].

NDTs serve as sophisticated virtual replicas of network setups and operational strategies, pivotal in enhancing real-time monitoring, predictive analytics, and pre-implementation simulations of network changes. Research on DTs in the network has just begun, and its application is still in the infancy stage [15], particularly in the context of 6G networks. Most of the cases and studies that claim to introduce NDTs in networks, such as 5G, are partial NDTs, for example, focusing on the RF part or the network management part [7]. Therefore, there is a lack of full-stack NDTs and overall architecture, especially for the next-generation 6G network. To address these gaps, we propose a high-level architectural framework for 6G Network Digital Twins system, consisting of three key layers: the Physical Twin Layer, Digital Twin Layer, and Application and Service Layer. We also explore potential strategies for deploying 6G NDTs and emphasize the importance of selecting appropriate tools for their implementation.

The remainder of the paper is organized as follows. Section 2, reviews existing literature, highlighting the fragmented nature of current NDT studies. Section 3 introduces the proposed architecture for a 6G NDT. Section 4 benchmarks existing software tools that can be used for building NDTs, looking into aspects of scalability, performance, and extensibility, and then provides recommendations for future tool selection and architecture refinement, while Section 5 concludes the paper.

2 Related Work

It is hard to have a unified and standard structure for an NDT system[8]. Currently, the three-layer or four-layer architecture is relatively common in the academic [1, 12–14], but there is less consensus on the details and structure of each layer.

In [5], the basic idea of NDTs was retained. Ericsson implemented a 5G NDT, providing a secure virtual environment for testing and optimizing parameters like radiated power, with visualization through Nvidia Omniverse. They enhanced network performance and power efficiency using reinforcement learning. The design effectively embodies NDTs' ideas and structure, utilizing AI as a tool. HEAVY.AI has developed an NDT application with a three-layer framework called HeavyRF [7], which integrates the SQL backend of Omniverse and HeavyDB, and is able to run real-time RF simulations on extremely high-resolution terrain data. Additionally, the study by Demir et al. [3] integrated vision from the outset, optimizing beam selection by extracting the spatial distribution of scatterers affecting wireless propagation from user-side camera images. These studies focus primarily on network parameter optimization. A more comprehensive approach can be seen in CAVIAR-related research [1], which introduced an all-in-loop co-simulation that includes 3D space and vision, incorporating the essential components of a digital twins system, which can be considered a simplified prototype.

However, most of these studies primarily focus on a part, like radio, with relatively little attention given to the digital twinning of other parts of the network, in other words, full stack 6G. Therefore, our work focuses on the full stack 6G. We explore a modular approach to the architecture, discussing the functional divisions and components, and based on this structure, we investigate the principles and strategies for implementing a more comprehensive full-stack 6G network digital twin system.

3 High-level 6G NDTs architecture

We define NDT as: *a dynamic, virtual replica of a physical network's infrastructure, operations, and lifecycle. It mirrors, simulates, and interacts with the physical network in real-time, enabling continuous synchronization between the digital and physical environments. [6, 9]. It uses data-driven computational models to maintain real-time conditions while forecasting future states [7]. An NDT features bidirectional interfaces that not only update the digital twin based on changes in the physical network but also allow the digital twin to issue commands to alter the physical network.*

In this section, we proposed a three-layer general 6G NDTs system structure, including the Physical Twin Layer, Digital Twin Layer, and Application and Service Layer, as shown in Figure 1.

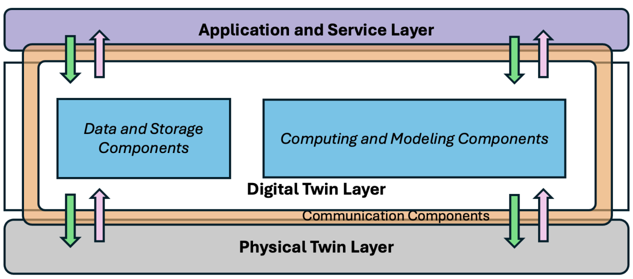

Figure 1: 6G network digital twins high-level architecture

3.1 Physical Twin Layer

The physical twin layer is primarily responsible for gathering and initially processing data from the physical environment.

In terms of composition requirements, the principle is that all objects in the physical world that can be modeled as digital models should be involved [11]. Network-related hardware devices, such as User Equipment (UE), Next Generation NodeB (gNodeB), and other hardware within the Radio Access Network (RAN), will inevitably be included at this layer. However, to achieve a more comprehensive full-stack digital twins system, environmental information must also be integrated, which is often ignored. For instance, location data is essential for positioning, along with other factors such as physical surfaces, as they affect signal attenuation.

In terms of performance requirements, the physical twin layer of a 6G NDTs system must have hardware capable of supporting high-frequency, efficient, and reliable data synchronization to ensure seamless updates and command execution between physical and digital components. We refer to the interfaces required for such communication as Southbound Interfaces (SBIs), which primarily handle physical-to-virtual and virtual-to-physical communication. The devices within the physical twin layer should also support physical-to-physical communication. Clearly, the design of 6G physical networks must meet fundamental requirements to support ultra-high data rates, low latency, and massive connectivity. Moreover, energy efficiency, security, and seamless interoperability with existing networks are also critical components.

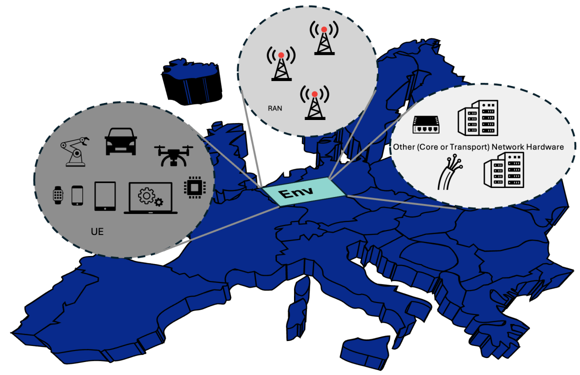

Figure 2: Physical twin layer

3.2 Digital Twin Layer

The Digital Twin Layer includes digital counterparts of physical objects but is not limited to these representations. In functional components, we propose that this layer should minimally include three core components: Communication Components, Data Storage Components, and Computing and Modeling Components. Data storage and computing can both be distributed and centralized. Security should be considered in every function. To simplify, we will not abstract the security components separately for now.

3.2.1 ***Communication Components.*** They act as the essential link between physical and virtual objects and virtual-to-virtual communication. They manage the uninterrupted transmission and transformation of data, including instructions and influences to manipulate the physical network. This module converts raw data into structured formats that the Virtual Twins Layer can readily use, optimizing compatibility and usability across the system. It also integrates advanced data processing functions such as mapping to align real-world elements with their digital counterparts.

3.2.2 ***Data Storage Components.*** The Data Storage Components are for effective data management, encompassing temporary storage for quick data processing, metadata storage for efficient organization and retrieval, and robust data security measures. Data can come from various sources. For example, real-world data can be obtained from the physical world in real-time, while simulated data can also be cleaned, reorganized, stored, and packaged into datasets to support future analysis and use or training AI models. Together, these components should also be under a comprehensive system that safeguards data integrity while facilitating smooth and secure data operations. Storage can ideally be distributed, to fit the 6G architecture like D6G, where each network device can run a simplified 6G stack [10]. While centralized storage is also an option, it may lead to data ownership and security issues, despite being more straightforward for data modeling and computational tasks.

3.2.3 ***Computing and Modeling Components.*** In terms of composition, this module is mainly composed of two parts:

- Virtualized Hardware: This part involves the virtualized hardware. The choice between simulation and emulation is based on specific requirements. If the network device is

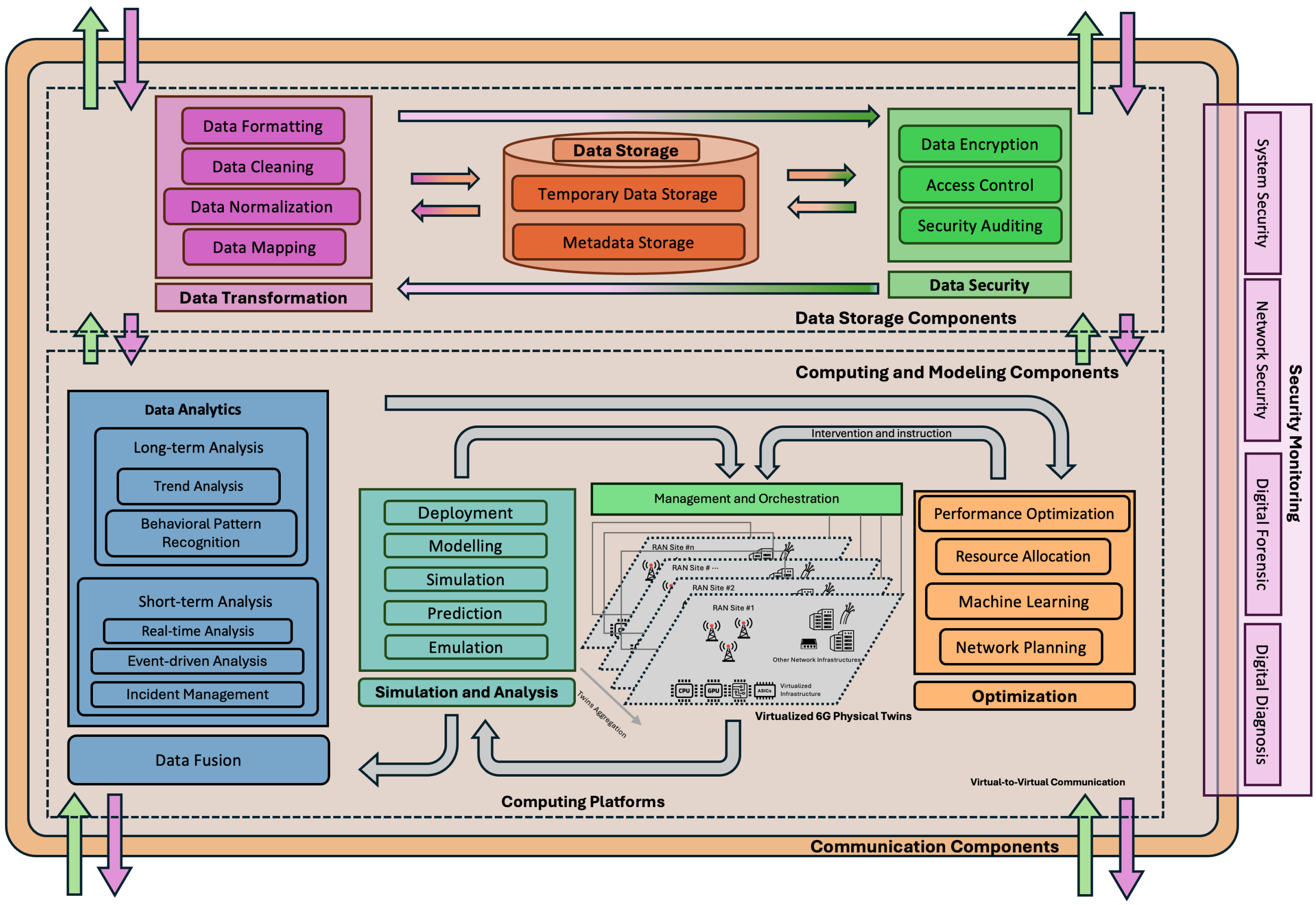

Figure 3: Digital twin layer

required to run a 6G stack internally, emulation should be used as much as possible. We will have a deeper discussion of criteria in Section 4.1.

- Containerized software: This part involves the implementation of service and network functions as software modules. With the introduction of SDN, software components enable dynamic and flexible network management by separating the control plane from the data plane. This allows centralized control of network traffic and resources, and can be programmed to adjust in real-time to meet specific requirements, such as load balancing or Quality of Service (QoS) optimization. The management component is not abstracted separately since it should also be included in the software part.

In terms of function, The Computing and Modeling Components should be capable of abstracting and aggregating computing and modeling units of various complexities and granularity to meet specific requirements. This includes capabilities for modeling, simulation, and prediction, as well as conducting 'what-if' analyses. Additionally, it should incorporate a comprehensive fault detection system capable of swiftly identifying and diagnosing operational issues alongside robust system optimization tools that fine-tune performance across diverse network scenarios. Moreover, its predictive maintenance algorithms proactively suggest repairs and upgrades, significantly reducing downtime and extending the lifespan of network components. To be noted, These advanced functionalities often rely on deep analytics and machine learning algorithms, which substantially support the upper Application and Service Layer. These functions should ideally operate like stateless services, generating relevant storage and logs, which can be automatically saved to external storage. This data can then be combined with data generated by the network itself, such as telemetry data, to contribute to model training. AI models can replace traditional data modeling methods and be applied to network management, such as resource allocation, congestion control, and network slicing [4]. These algorithms can continuously learn from ongoing operations and simulations, thereby enhancing their accuracy and efficacy.

3.3 Application and Service Layer

The Application and Service Layer is the uppermost tier of the 6G NDTs system architecture, where it directly orchestrates services and delivers enhanced experiences for end-users.

This layer incorporates various applications spanning augmented and virtual reality, autonomous vehicles, smart city technologies, and tele-medicine, all designed to seamlessly integrate with daily human activities and infrastructural operations. This layer can communicate with the digital twins layer, which we call interfaces between them Northbound Interfaces (NBI). Moreover, the Applications and Service Layer should be characterized by its Service-oriented Architecture (SOA) which enables it to offer modularized and reusable services. This modularity allows for services to be independently deployed, managed, and scaled, meeting specific user demands without affecting the overall system. Furthermore, it supports isolated operations where individual services can function in a standalone mode, enhancing fault tolerance and reducing dependencies. This isolation is essential for ensuring that any disruptions in one service do not cascade to others, thereby maintaining the robustness and continuity of user services. Besides, each service should be both resilient and precisely tailored to meet evolving user needs and environmental contexts.

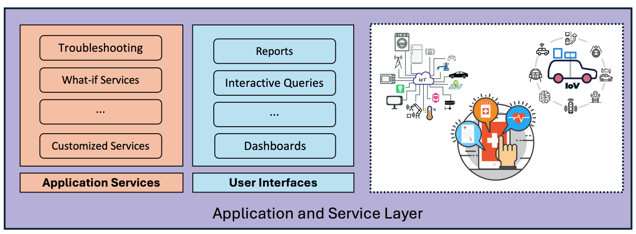

Figure 4: Application and service layer

4 Benchmarking Software Tools towards NDTs

In our 6G NDTs system architecture, the digital twin layer plays the most important role, and the digital network is its core. Therefore, it is essential to prioritize the capability to digitally replicate and operate the physical network, which places specific demands on the research and usage of network simulators or emulators.

A network simulator models the behavior of a network, and a network emulator replicates the actual environment for real-time testing. Emulators provide a more accurate and realistic representation of network behavior by replicating hardware and software conditions. While a network simulator/emulator can provide a robust foundation for creating NDTs, it needs to be integrated with real-time data, continuously updating the model and validating its accuracy against the real network. Meanwhile, it should also be matched with sufficient scalability and maintain a balance between performance overhead and simulation/emulation fidelity.

Nevertheless, scalability and extensibility are challenging to quantify. Unlike 5G, the absence of a standardized 6G architecture requires current tools to have sufficient extensibility to serve as a foundation for development. The complexity and novelty of these features demand simulators and emulators that can accurately replicate the physical and logical behaviors of 6G networks, while also being easily extended and updated as 6G standards evolve. The performance overhead on the network common parts, on the other hand, can be relatively compared to obtain a general estimate. Thus, an experiment can be conducted.

4.1 Resource Performance

Simulators and emulators have varying performance overheads characteristics, as differing implementation principles. Some tools, like GNS3, utilize QEMU for hardware emulation, which allows for more accurate modeling of network behaviors by closely mimicking real hardware. On the other hand, some simulators focus solely on abstracting the network environment without detailed hardware emulation, which can result in lower resource consumption but might not capture the full complexity of real-world networks.

In order to briefly verify the differences that different kinds of tool may bring, we conduct experiments to compare the resource utilization of different network simulators/emulators under the same topology deployments and functions in a containerized environment. The choices include NS-3, GNS3, and Mininet, as they are representative of our purpose. Among them, NS-3 is a simulator in C++ that simulates the behavior of network protocols through discrete events. Mininet and GNS3 are emulators, developed by Python, but one uses network namespace and virtual Ethernet interface to create topologies, and the other uses virtualization technology such as QEMU, or Virtual Box runs the actual network operating system. The comparison focused on how the computing resource changes when simulators/emulators handle dynamic changes in network scale and topology.

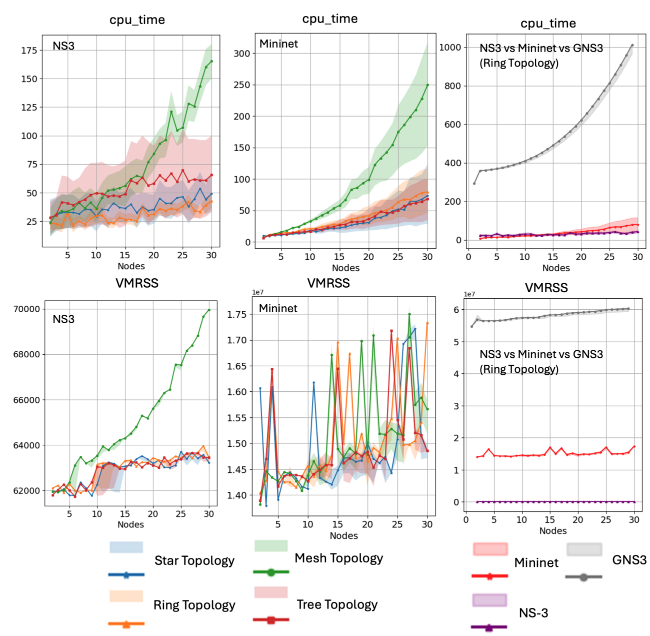

Figure 5: Simulators/emulators benchmark

Experiments show that the performance usage of different simulators/emulators varies greatly, as shown in Figure 5. In general, as the network scale increases, resource consumption also grows. Additionally, simulating/emulating more complex network topologies, such as fully connected networks, requires more CPU and memory. By conducting a horizontal comparison of different types of network simulators, we can find that using C++ has better results and more stable usage of resources, especially in memory. The CPU overhead of emulating can be several times that of simulating, and it also consumes significantly more memory. At the same time, emulators such as GNS3 also have higher overall usage due to emulating real IOS images and, due to this feature, come with higher scalability, even though the hardware requirements are the highest.

4.2 Tools Selection

The concepts of NFV and SDN are applicable to both 5G and 6G. These features give possible ideas and references for choosing appropriate network tools to build a full-stack 6G NDTs system. Since the physical twin layer mainly consists of actual network infrastructure, and the application and service layer is more like the interface exposed to higher layers, specific requirements can be addressed like the way of software development. Therefore, they are not our focus. Instead, we should pay more attention to the digital twin layer.

In our proposed DT layer, there are three components. The computing and modeling are based on digital network infrastructure, which should include RAN, core network (called in 5G), transport network, and the physical environment. In network infrastructure, the main simulation challenges lie in the RAN and core networks, with the transport network requiring relatively fewer updates. Various open-source projects, such as free5GC and Open5GS, are available for 5G core networks, but they differ in functionality and require careful selection. Some projects focus specifically on RAN, such as UERANSIM. Due to NS-3's excellent extensibility, some extensions based on it exist, such as 5G-LENA and ns3-mmwave. Now, combining free5GC with UERASIM has shown to be a successful approach in 5G. Some modules can be reused in 6G, such as expanding millimeter-wave modules to higher frequencies or terahertz bands. However, the core network may rely on new open-source projects and tools.

For simulating the high-fidelity physical environment, Unity3D and Unreal Engine are common choices in not only wireless networks', but also other DTs. Tools like Nvidia Omniverse can also be promising options. The interfaces between the physical and digital world have the greatest challenges and design complexities in the communication components. To implement and maintain highly efficient synchronization, one can either investigate adapting existing technologies like MQTT, Kafka, RabbitMQ, and OPC UA or develop specialized tools specifically for this purpose. Meanwhile, the data storage components should be able to retain time series data and use Time Series Database in combination with distributed file storage. In addition, tools with special structures such as graph databases can also be considered.

Like 5G, 6G should also at least meet the key requirements mentioned above for fully deploying digital twins. The extreme KPI feature of 6G also makes this a greater difficulty. Given that 6G NDTs are complex systems with numerous services and applications, and AI/ML interfaces should also be distributed integrated, it is essential to consider a combination of multiple tools, ensuring sufficient extensibility and interfaces.

5 Conclusion

In this paper, we have explored the architectural framework for implementing NDTs within the context of 6G networks. We proposed a three-layer architecture where each is designed to address the unique challenges of replicating and managing the next-generation mobile network environment. Through benchmarking different simulation and emulation tools, we demonstrated the impact of tool selection on system performance, explaining the need for careful consideration when choosing technologies for NDTs system deployment.

As 6G technology continues to evolve, future work will focus on refining the proposed architecture, particularly in finding specialized interfaces for seamless synchronization between physical and digital layers. Additionally, we will explore integrating advanced AI-driven analytics and federated learning techniques to enhance the predictive capabilities and safety of 6G NDTs systems.

6 ACKNOWLEDGMENTS

This work is funded by the Dutch 6G flagship project "Future Network Services" (National Growth Fund).

References

[1] João Borges, Felipe Bastos, Ilan Correa, Pedro Batista, and Aldebaro Klautau. 2024. CAVIAR: Co-Simulation of 6G Communications, 3D Scenarios and AI for Digital Twins. *IEEE Internet of Things Journal* (2024), 1–1.

[2] Shuping Dang, Osama Amin, Basem Shihada, and Mohamed-Slim Alouini. 2020. What should 6G be? *Nature Electronics* 3, 1 (2020), 20–29.

[3] Utku Demir, Suyash Pradhan, Richard Kumahia, Debashri Roy, Kaushik Chowdhury, et al. 2023. Digital twins for maintaining QoS in programmable vehicular networks. *IEEE Network* 37, 4 (2023), 208–214.

[4] Cyril Shih-Huan Hsu, Danny De Vleeschauwer, and Chrysa Papagianni. 2023. SLA Decomposition for Network Slicing: A Deep Neural Network Approach. *IEEE Networking Letters* (2023).

[5] Richard Kerris. 2021. Ericsson Builds Digital Twins for 5G Networks in NVIDIA Omniverse. https://blogs.nvidia.com/blog/ericsson-digital-twins-omniverse/

[6] Latif U. Khan, Walid Saad, Dusit Niyato, Zhu Han, and Choong Seon Hong. 2022. Digital-Twin-Enabled 6G: Vision, Architectural Trends, and Future Directions. *IEEE Communications Magazine* 60, 1 (Jan. 2022), 74–80.

[7] Xingqin Lin, Lopamudra Kundu, Chris Dick, Emeka Obiodu, Todd Mostak, and Mike Flaxman. 2023. 6G Digital Twin Networks: From Theory to Practice. *IEEE Communications Magazine* 61, 11 (Nov. 2023), 72–78.

[8] Qing Liu, Bin Liu, Guan Wang, and Chen Zhang. 2019. A comparative study on digital twin models. In *AIP conference proceedings*, Vol. 2073. AIP Publishing.

[9] Huan X. Nguyen, Ramona Trestian, Duc To, and Mallik Tatipamula. 2021. Digital Twin for 5G and Beyond. *IEEE Communications Magazine* 59, 2 (Feb. 2021), 10–15. Conference Name: IEEE Communications Magazine.

[10] Gergely Pongrácz, Attila Mihály, István Gódor, Sándor Laki, Anastasios Nanos, and Chrysa Papagianni. 2023. Towards extreme network KPIs with programmability in 6G. In *Proceedings of the Twenty-fourth International Symposium on Theory, Algorithmic Foundations, and Protocol Design for Mobile Networks and Mobile Computing*. 340–345.

[11] Wen Sun, Wenqiang Ma, Yu Zhou, and Yan Zhang. 2022. An Introduction to Digital Twin Standards. *GetMobile: Mobile Computing and Communications* 26, 3 (Oct. 2022), 16–22.

[12] Fengxiao Tang, Xuehan Chen, Tiago Koketsu Rodrigues, Ming Zhao, and Nei Kato. 2022. Survey on Digital Twin Edge Networks (DITEN) Toward 6G. *IEEE Open Journal of the Communications Society* 3 (2022), 1360–1381. Conference Name: IEEE Open Journal of the Communications Society.

[13] Kunlun Wang, Yongyi Tang, Trung Q. Duong, Saeed R. Khosravirad, Octavia A. Dobre, and George K. Karagiannidis. 2023. Multi-Tier Computing-Enabled Digital Twin in 6G Networks. arXiv:2312.16999 [cs].

[14] Xiaoyun Wang, Chao Liu, Jiwei He, Nanxiang Shi, Tong Zhang, and Xiaohan Pan. 2024. 6G Network Architecture Based on Digital Twin: Modeling, Evaluation, and Optimization. *IEEE Network* 38, 1 (Jan. 2024), 15–21. https://doi.org/10.1109/MNET.2023.3333822 Conference Name: IEEE Network.

[15] Yu Zheng, Sen Yang, and Huanchong Cheng. 2019. An application framework of digital twin and its case study. *Journal of Ambient Intelligence and Humanized Computing* 10, 3 (March 2019), 1141–1153.