IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 31 December 2024, accepted 15 January 2025, date of publication 20 January 2025, date of current version 27 January 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3531947

SURVEY

Network Digital Twin Toward Networking, Telecommunications, and Traffic Engineering: A Survey

REZA POORZARE[1], (Member, IEEE), DIMITRIS N. KANELLOPOULOS[2], VARUN KUMAR SHARMA[3], POULAMI DALAPATI[3], (Member, IEEE), AND OLIVER P. WALDHORST[1], (Member, IEEE)

[1]Institute of Data-Centric Software Systems (IDSS), Karlsruhe University of Applied Sciences, 76133 Karlsruhe, Germany
[2]Department of Mathematics, University of Patras, 265 00 Patras, Greece
[3]Department of Computer Science and Engineering, The LNM Institute of Information Technology, Jaipur 302031, India

Corresponding author: Reza Poorzare (reza.poorzare@h-ka.de)

This work was supported by the Ministry of Science, Research, and the Arts Baden-Württemberg (MWK) under Project bwNET2.0.

ABSTRACT Network Digital Twin (NDT) is an evolving technology that provides a framework through which a network administrator can have a virtual representation of a computer network. As a result, analysis, monitoring, testing, running new protocols, and more can be performed using the NDT before the final deployment of the developed approach. In this way, the consequences of direct deployment and the negative impact on network operations can be avoided. Telecommunications, along with traffic engineering as one of its critical components, play a prominent role across various networking domains, including Internet service providers, data centers, cellular networks, intelligent transportation systems, and smart cities. In this context, NDT has the potential to serve as a key enabler for optimizing these domains by providing a digital framework, which can facilitate the evaluation and enhancement of different scenarios. Accordingly, this paper presents a comprehensive survey on how NDT can facilitate advancements in network traffic engineering across a wide range of networking domains. First, we start with an in-depth analysis of the evolution of the network digital twin technology and provide a comparison with simulation tools. Next, we examine the role of NDT in various networking and telecommunication domains. We also explore the applicability of NDT technology from a traffic engineering perspective across different network types. Subsequently, we highlight key open research questions and potential future directions that warrant further investigation. Finally, we conclude by outlining the promising future trajectory of NDT within the aforementioned domains.

INDEX TERMS Digital twin, network digital twin, telecommunications, traffic engineering, traffic optimization.

I. INTRODUCTION

Network Traffic Engineering (TE) is one of the critical pillars of high performance for computer networks within various telecommunications domains, as it strives to adjust the traffic flows in a way that reduces congestion, latency, and packet loss. It is also an enabler for improving the user experience and delivering particular services with Quality of Service (QoS) [6], [16]. These goals can be achieved by using mechanisms to route network traffic efficiently through switches and routers in a way that maximizes resource utilization by establishing proper load balancing on network devices [6]. Consequently, traffic engineering is a fundamental aspect of telecommunications and plays a critical role across various networking domains.

The associate editor coordinating the review of this manuscript and approving it for publication was Tariq Umer.

Improving telecommunications performance and reliability is highly dependent on having the right traffic engineering approaches, which can be achieved by providing data, voice, and video services in a way that ensures a high user experience through optimal communication without delays, interruptions, or degradation of connection quality. However, traffic engineering solutions encounter issues and experience difficulties in attaining the above objectives in various networking and telecommunications domains, such as cellular communication [271], [292]. In addition, networks are expanding at a rapid pace, and new data-hungry applications such as holograms and augmented reality have emerged, resulting in high data rates and low latency demands that make routing even more challenging [154], [181]. Moreover, networking and telecommunications include a vast range of domains, including cellular networks, the Industrial Internet of Things (IIoT), intelligent transportation systems, smart cities, etc., each of which has its own unique characteristics.

In addition to the complexity that networks present as a hurdle to having proper routing mechanisms, testing and approving their functionality are also other challenges. It is almost impossible to create a new routing algorithm targeting a particular telecommunication field and test it on a running network without experiencing consequences, such as service availability interruptions. On the one hand, network simulation and emulation tools can be used to analyze the new algorithms. On the other hand, these tools suffer from many shortcomings, such as accuracy, speed, and scalability, which make it difficult to represent a highly accurate copy of the network and obtain reliable results. Moreover, working with simulation tools has its own challenges, such as gaining knowledge of the particular tool, and they do not cover all domains of telecommunications, such as Six Generation (6G) [209]. Nevertheless, it is important to test and analyze new approaches prior to implementation, and it is not advisable to implement a new traffic engineering algorithm in a specific telecommunications domain such as 6G without pre-implementation experiments. Therefore, we need a tool that goes beyond simulations and covers a wide range of networking and telecommunications domains.

Among the trending technologies for both academia and industry, Digital Twin (DT) [60] has gained massive interest in recent years. In short, a DT is a digital clone of a physical object that can mimic and emulate its behavior. In other words, it can be a digital representation of a real system [227]. The network version of the DT, called Network Digital Twin (NDT), is a virtual replica of a physical network, enabling simulation, analysis, and optimization of network operations, management, and performance, by mimicking the dynamic behavior of the physical network. The NDT provides a framework through which administrators/operators can have a digital representation of the network with real-time synchronization of the state of the physical network and the twin. As a result, analysis, monitoring, testing, running new protocols, and more can be performed on the NDT prior to its implementation [27]. This approach can avoid the consequences of direct implementation and prevent negative impacts on a network's operations. Furthermore, NDT makes it feasible to test and analyze conditions that may rarely occur in the network [11], [273].

Despite the fact that NDT is an evolving technology, broad-successful research could be accomplished in various domains, such as Fifth Generation (5G) [54], 6G [17], IIoT [139], network traffic prediction [165], and Software Defined Networking (SDN)-based networks [5], [189]. Moreover, there are many surveys in the field, such as [21], [81], [130], [273], which cover various scenarios and use cases of DT and NDT. However, there is a lack of a survey that analyzes NDT in different areas of networking, telecommunications, and their main pillar, i.e., traffic engineering, and how NDT can be an enabler to improve traffic engineering in different areas of telecommunications. So, the main goal of this paper is to fill the mentioned gap. As a result, the contribution of the paper will be as follows:

- An investigation of the key technologies of NDT and their standpoint within the field of networking and telecommunications.
- A comparison of NDT and simulation tools to find out their similarities, advantages, and disadvantages.
- A thorough analysis of network traffic engineering, including a detailed review of existing algorithms and an investigation of related NDT initiatives.
- The investigation of the network digital twin in various areas of telecommunications where traffic engineering can play a significant role.

The rest of the paper is organized as follows: Sections II and III are a tutorial on NDT. In particular, Section II provides the evolution path for the NDT concept while Section III discusses NDT as a network replication enabler. Section IV presents important surveys related to NDT. Section V presents the review methodology we used to write this article. Section VI presents key technologies in NDT, while Section VII presents technical challenges that need to be addressed. Section VIII provides a comparison of NDT and simulation tools. The application of NDT in various areas of telecommunications is discussed in Section IX. Section X presents how Machine Learning (ML) and Deep Learning (DL) techniques can be integrated with other technologies to improve the capabilities of NDTs. Section XI elaborates on network traffic engineering and NDT initiatives. Section XII discusses open research challenges, while Section XIII presents future research directions. Finally, Section XIV concludes the paper. The sections and the issues addressed in this review paper are depicted in Figure 1.

If the reader is familiar with the concept of NDT and the basic modeling and training processes of an NDT, s/he may start directly from Section IV.

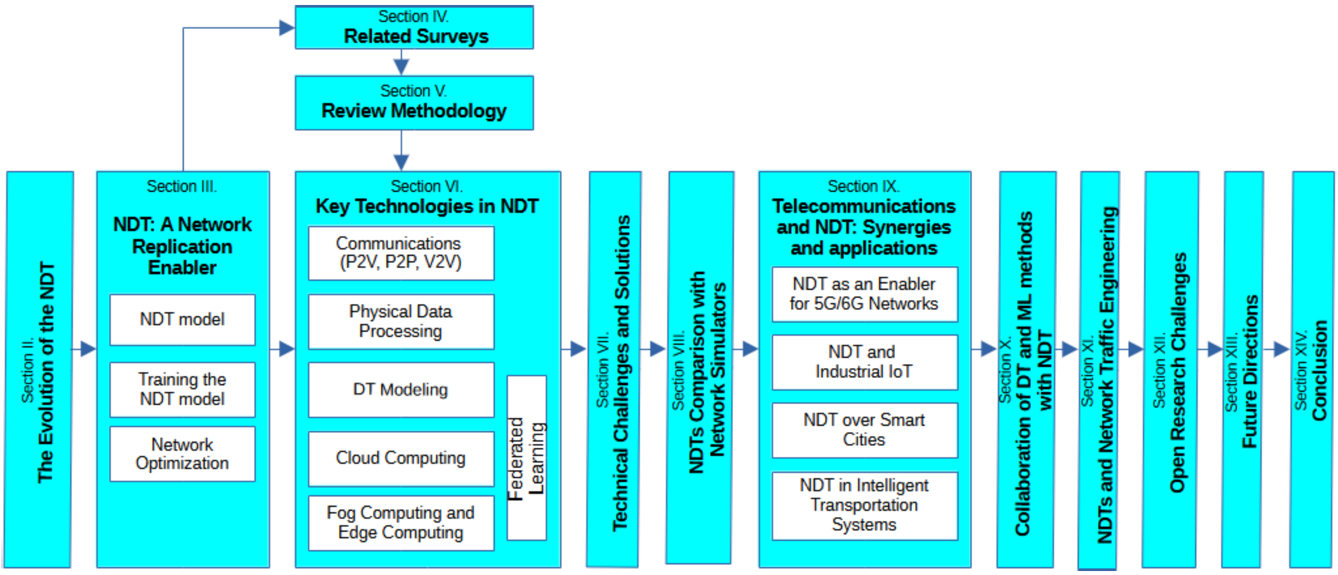

FIGURE 1. Paper structure.

II. EVOLUTION OF THE NETWORK DIGITAL TWIN

In the current Internet era, it has been assessed that users equipped with sophisticated handheld devices generate approximately 403 million terabytes of data every day globally. Handling such huge amounts of data is beyond human capability. Recent breakthroughs in Artificial Intelligence (AI), big data processing, and ML have opened new avenues for handling such vast data. Plus, with the integration of DT into Industry 4.0, we are witnessing a tremendous advancement in DT technology, mainly in smart city applications, aviation, manufacturing units, and healthcare sectors.

The spread of smart devices with Internet of Things (IoT) capabilities impacts how data is shared between various sources. Essential application (big) data is continuously sensed, collected, and communicated over the Internet by smart sensors and actuators via Wireless Sensor Networks (WSNs) [99], [217]. Such massive amounts of data are not one-dimensional or homogeneous in nature; they are rather multi-dimensional and heterogeneous and necessitate huge amounts of storage space as well as far more sophisticated processing techniques. The development of cloud computing and multi-dimensional big data processing, as well as the handling of heterogeneous multi-dimensional data procedures, and new data aggregation practices, all made it feasible to analyze colossal amounts of data to exploit the most vital information possible for enhancing the system's performance.

The combination of big data analytics with virtual AI-based models (for describing physical objects) raised advanced options that we had never thought of previously. One promising approach is the DT concept [21], which is *the digital replica of a real-world object that could be a physical system, a machine, a computer, a device, or a process* with an important advantage over digital models; the real-time synchronization with the physical entity or process.

The concept of a DT is discussed at different levels of expertise. For instance, the authors [212] pondered the end product as a DT, while a few authors [68] considered it as the complete product lifespan. Additionally, the authors in [164], [215] came up with several debatable issues, such as sustainability, privacy, and security (particularly for IoT devices), power-related issues, among others, and offered their insightful opinions to address them.

According to [253], a DT is a virtual depiction of a device, or some set of devices, or a complete system. The DT uses large amounts of data and mainly processes it using several ML- or AI-based pattern recognition algorithms [193]. Then, DT produces vital information that aids in any field's innovation, optimization, and release of new applications [159].

Recently, the focus of DTs has evolved substantially from being just digital mimics of immobile physical items to dynamic planning, learning, reasoning, and future advancement. As discussed in [164], modern DTs not only reflect and display real-time static physical entities, but also complex systems.

Nowadays, many nations are attempting to deploy twin-based systems [104]. By 2026, system DTs will hold the largest market share of all types [52]. To achieve this goal, standardization efforts have started focusing on NDT, the networking version of DT. Hereafter, we briefly discuss the standardization efforts achieved by the Internet Engineering Task Force (IETF) and the International Telecommunication Union Telecommunication Standardization Sector (ITU-T).

In the architecture suggested by IETF for NDT [300], a DT network comprises four key building blocks: data, models, interfaces, and mapping. Different network elements, such as controllers, routers, middleboxes, physical and virtual appliances, etc., collect a variety of data that represents network topology, configuration, operational state, trace data, metrics, and process data. Next, this data is accumulated within the data storage area to supply appropriate and precise data service support to construct DT models. The DT platform uses this data to generate the states of both real and virtualized network instances. The previously built DT is frequently updated by using new data and a variety of models (e.g., knowledge graphs, dataset models, service models, and data models) can be combined to create a high-level abstraction that can be used to deduce reasoning data. The DT can use this reasoning data to sustain a range of network applications. Next, interfaces make sure that the NDT can operate with the business apps that are currently in use. To facilitate real-time data gathering and control on the real network, standardized southbound and northbound interfaces are required. One-to-one and one-to-many mappings are required to create a real-time interactive association between the real network and the twin network.

ITU-T [56] makes available an NDT architecture for mobile and telecommunication networks and specifies its requirements. The ITU-T has begun work on the recommendation ITU-T Y.3090 [194], which outlines the architecture, functional requirements, and service requirements for twining a real network with a DT. One important feature of this architecture is its open interfaces (northbound and southbound). The northbound interfaces exchange data between the network applications and the NDT, while the southbound interfaces link the virtual network and the real network. This architecture/system can visualize and monitor the status of the physical network, recover vital network data effectively, and explore network applications (e.g., network management and optimization) through user-friendly, interactive interfaces. The NDT architecture (Figure 2) [194] has three layers: network application layer, NDT layer, and physical network infrastructure layer. Each layer corresponds to a specific domain (i.e., applications, NDT, physical network) and a double closed loop. The NDT layer consists of the unified data model, the unified data repository, and the DT entity management counterpart. The core element of the NDT system is its functional model, which performs traffic analysis, fault detection, network planning, scheduling optimization, etc. The management of the DT entity focuses not only on the DT model but also on the topology and network security.

III. NDT: A NETWORK REPLICATION ENABLER

NDTs are a novel category of tools for network modeling that mainly use ML approaches to create precise data-driven digital network representations [194]. An NDT is a framework formed by linking numerous DT nodes that communicate with each other. NDT enables the synchronized evolution and dynamic interaction of physical objects and virtual twins. Real objects and virtual twins can share information, cooperate, communicate, and finish tasks together.

The NDT not only affects network functions in real-time but also changes with the actual network over its existence by evaluating past and present work conditions, identifying possible problems, projecting future performance, and assisting in decision-making regarding network deployment, operations, and other matters.

To build the model of an NDT, a training process is executed based on collected data from the real network. To enable the learning process in the NDT model through data collection, dedicated network testbeds, real-world network data, or network simulators can be employed to gather data. These datasets are used to train data-driven digital network models. The diversity of this data should allow the network operator to replicate a large variety of possible scenarios, such as different levels of congestion and connection failures, by leveraging sophisticated DL techniques [41], [66].

Common functionalities that benefit from NDTs for efficient network management and control are [11]: *Troubleshooting*, *What-If-Analysis*, *Network Planning*, and *Anomaly Detection*. For example, many NDT services provided to network operators, such as predictive maintenance and system condition monitoring, are based on anomaly detection and monitoring enabled by ML and DL techniques.

A. NETWORK DIGITAL TWIN MODEL

To build an NDT of a communication network, basic topology models and network properties are employed for deciding the configuration of the network properties, operation data, connection status, network topology links, and other relevant network information. Virtual network properties assist in verifying the functionality of the data and control planes since they can carry out packet processing through network simulators.

An NDT model uses different tools to train and validate the model of the real network. In terms of small-scale networks, basic network models can be created by using network emulators (like EVE-NG [59] and GNS-3 [72]) or network simulators (like MININET, network simulator 2 (NS-2), and network simulator 3 (NS-3)). However, network emulators and network simulators have several disadvantages, including high consumption of resources, limited performance analysis capability, and inadequate scalability. As a result, network modeling can also be obtained with other tools, including fluid, analytical, and function models [281]. However, they also suffer from some flaws in terms of accuracy [281]. All these tools were shown to be highly inefficient for handling real-time networking scenarios, where enormous amounts of data are continuously created. Therefore, they are unable to meet the QoS and Quality of Experience (QoE) requirements imposed by the network users.

Further mathematical abstraction schemes were employed for designing fundamental network models. For instance, Giralt et al. [201], [202] introduced the Quantitative Theory of Bottleneck Ordering and Structure (QTBOS), which is a mathematical framework. Where there is no interference

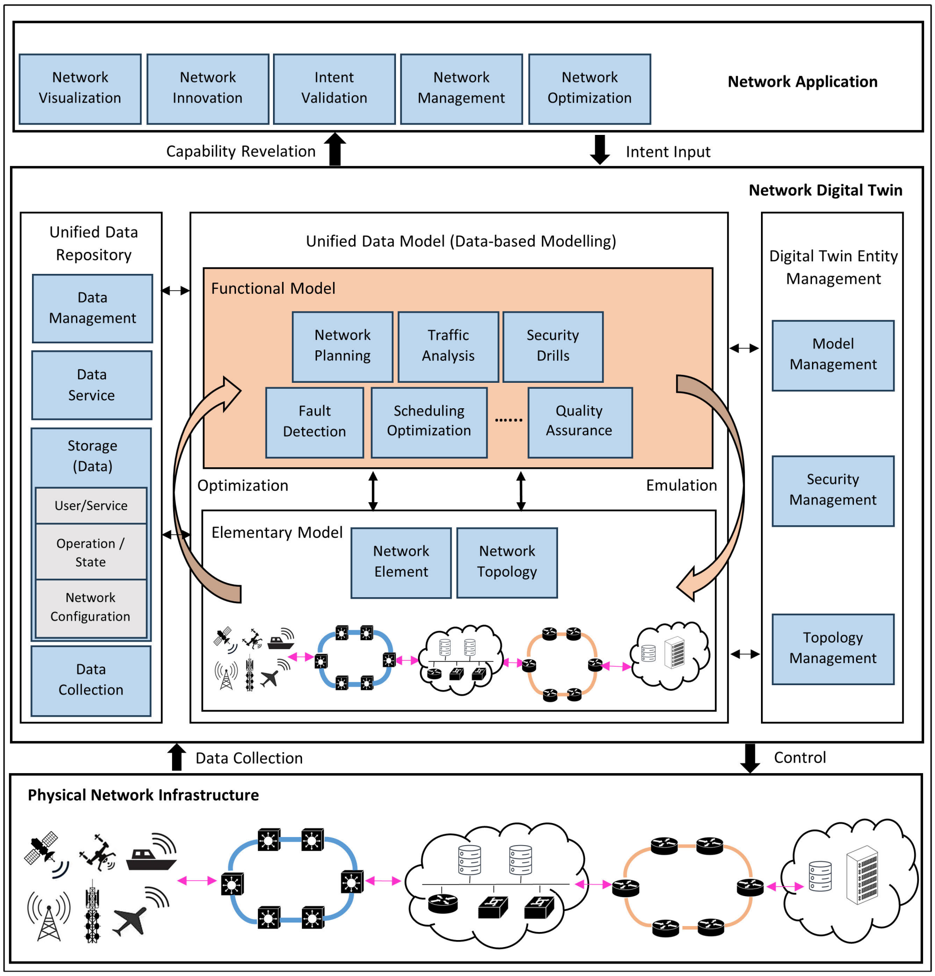

FIGURE 2. Reference architecture of an NDT.

in the traffic pattern, there should exist an optimum design that minimizes the Flow Completion Time (FCT), and maximizes throughput performance. The authors presented an approach of employing the QTBOS framework to design a network-based mathematical model. The fundamental component of QTBOS is a computational graph, known as "*bottleneck structures*", which enables the measurement of the forces of relations between flows and bottleneck links. Such graphs indicate the underlying relationships that occur between application traffic flows and bottlenecks. So, they contribute to determining the ripple effects of perturbations. Since these perturbations can be visualized as mathematical derivatives of the networked system, bottleneck structures can be utilized for computing improved network configuration. Bottleneck structures can be employed to calculate performance metrics like average latency, and throughput. This idea may prove to be very useful for creating an NDT.

The functional models provide dynamism and motivation for evaluating network performance patterns that lead to important decision-making procedures. They often include data-driven AI/ML techniques. Especially, DL algorithms can improve the decision-making processes, by evaluating network parameters, which leads to the creation of effective network function models. Here, real-time data training allows data-driven AI/ML algorithms to easily predict and take into account the entire range of network features, helping them to achieve high levels of accuracy.

Many studies focused on using Neural Network (NN) to model communication networks [261]. These NN-based models [149], [204], [246] are used to realize network optimization. Some authors [155], [261] introduced the use of well-known NN-based architectures, which may not be feasible in a communication networking environment because those environments are represented by graphs, and this NN-based learning process over such graph-structured data is much complex in terms of proper training encompassing a huge dataset. Because of this, these models only provide restricted accuracy in controlled environments. Also, it is hard to generalize them to routing critical settings and changing network topologies. Rusek et al. [204] addressed this problem and utilized the notion of Graph Neural Network

(GNN) to model network design. They created an effective message-passing process to extract complicated relationships from routing settings and network topologies. Then, they used GNN to model the graph-structured data. They started using GNN for such scenarios since they are specially designed to comprehend the complicated connection between the graph elements [65].

An NDT can also be considered as a black box where a collection of models (mainly based on ML) will be applied [11]. Within the application's context, these ML-based models will be designed, implemented, and maintained within NDT accordingly. It produces network-related performance parameters like latency, packet losses, and channel utilization based on some input values, which are crucial network factors like network topology, traffic conditions, channel access, scheduling, forwarding, and routing schemes. Also, the tested inputs and the resulting outcomes are altogether application and context-dependent. The network operator can assess the resulting performance metrics outcomes corresponding to a modeled NDT in real-time scenarios by testing the set of input parameters to it. Simply, it is accomplished without requiring real-time network mimicking or executing costly and complex network simulations [11].

In industrial IoT networks, many network nodes are involved. Thus, the risk of network attacks is very high. In [64], a game-theory-based approach was proposed to analyze network security interactions between attackers and defenders. By managing the vulnerability lifecycle of DTs and optimizing security resource allocation, this study provides a strategic framework for protecting DTs and ensuring the secure development of IIoT. This approach not only mitigates risks but also supports the broader goals of modern digital economies and smart societies.

In [108], an architecture was proposed incorporating Timed Petri-Nets (TPNs) for IIoT network modeling within NDTs. By leveraging open-source tools, TPNs simulate both the state-driven behavior of the network and the influence of real-time conditions. Applications include fault detection and reliability prediction, with the Petri-net model capturing normal network behavior to facilitate real-time monitoring. Additionally, challenges in virtual–real mapping are addressed through system observability analysis. This approach validates the use of Petri-nets as a robust modeling framework, paving the way for more resilient and efficient IIoT networks.

Network simulators and NDT are perhaps the most appropriate paradigms for using digital representations of communication networks that help evaluate complex dynamic networking scenarios and, in turn, support the decision-making process. Considering their capabilities, each one has strengths and weaknesses and should be selected based on the requirements. The following sections will assist in selecting the most appropriate tool based on specific requirements.

B. NETWORK OPTIMIZATION

TE optimization problems, network planning, congestion minimization, load balancing, and detection of anomalies or faults in a physical network can be solved when NDT is joined with a network-optimizing entity. To do so, a set of network-related data is fed into the network model, which outputs a set of performance measures. For example, the network model predicts a specific routing scheme (configuration) for a certain per-path delay. At the same time, the optimization algorithm can produce a variety of routing schemes (configurations) that can be applied to achieve the desired per-path delay [65].

Figure 3 shows the network optimization process using the NDT. It is possible to execute the optimization process in a closed loop without requiring human involvement by using the following steps:

Step 1: The essential network requirements (or network intent) are specified by the network operator. The network operator may, for instance, state that the aim is to reduce the maximum link utilization for the real network. The network optimizer is notified of this crucial requirement.

Step 2: The network optimizer is in charge of establishing the optimal network configuration that satisfies the specified requirement.

Step 3: The network optimizer keeps searching for an acceptable solution until a halting (stopping) state is met. It continues searching if the network performance metrics (generated from the NDT) show that the answer/solution is insufficient.

Step 4: Lastly, the best solution (answer) discovered so far can be implemented straight into the real network. In **Step 5**, the real network may issue a new network event (e.g., a link breakdown) to the NDT.

A network optimizer must respond to the continuous changes in traffic, topology, and resource usage that occur in a dynamic physical network to provide effective network management. Notably, external factors may cause physical links to break, or the users of the real network may exhibit varying patterns of behavior that result in unpredictable surges in the usage of network resources. In this regard, Deep Reinforcement Learning (DRL) is a crucial technology as it has demonstrated excellent potential for effective network functioning in dynamic circumstances and complex scenarios [23], [41], [283]. However, DRL frequently yields less-than-ideal answers to intricate optimization issues. For example, it can be difficult to identify the ideal network settings (configurations) that maximize certain performance indicators in resource allocation situations. This is due to the possibility that there is a very wide solution space (or number of possible actions). Therefore, we need more thorough exploration techniques to identify the best solution. To increase the optimization performance, some works began merging DRL with other conventional optimization techniques such as Integer Linear Programming (ILP) [302]. Also, a new type of centralized optimization algorithm was suggested, which is based on the SDN paradigm [76], [154] to bring a centralized view of the network.

In dynamic networks (e.g., Mobile Ad-hoc Networks), network planning, congestion minimization, load balancing, and other TE problems can also be solved if a network optimizer is integrated with the NDT.

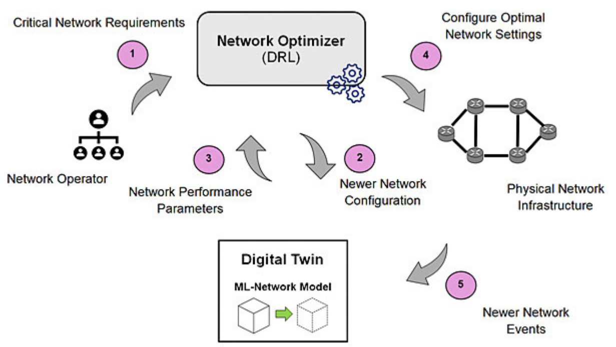

FIGURE 3. Network Optimization using NDT (adapted and redrawn from [11]).

C. TRAINING THE NDT MODEL

For modeling an NDT, the system needs many relevant datasets to be collected. In this regard, two issues need to be considered: the speed of model building and model training [230]. The main challenge in the model training part is how to convert the collected data into input for which the model can work. Such datasets can be collected from real-world scenarios, simulation tools, and/or from different non-production testbeds. Previous research [55], [69] proved that data collection from production environments is sometimes not feasible. Hence, current research focuses on producing training datasets in non-production scenarios, mainly with a dedicated testbed or network simulators. Another important challenge is to acquire diverse kinds of samples having various network characteristics for the generalization purpose [11].

The main challenge in training an NDT model with a large state space is not only the time taken in training but also maintaining the data accurately for further training. In most of the target problems, these states (populated with different data and features) need to be generalized using previous experiences which is the part of the training phase and can be gathered from past encounters with various states with similar problems. In ML [33], this issue is known as *generalization* [31], [230]. The key idea is how the large state space with huge datasets can be efficiently generalized to make an optimal decision by using limited amount of data, which are subsets of it. The performance measure of generalization is conducted upon a test set, an example dataset, which are different from the training dataset. During training, an NDT model partitions the data into three subsets: *training dataset*, *validation dataset*, and *test dataset* [33], [269]. The training dataset consists of the data upon which the model is trained; the validation set inputs the data to the model that are not included during the model training; and the test set is used to measure the performance of the model generalization. The reason behind using different datasets in generalization is that if the NDT model is trained on a test dataset, it will be difficult to check whether the model is correctly generalized or the model is making decisions just by memorizing the examples trained upon.

In a real-world network, environmental changes happen very quickly, and many unforeseen scenarios can occur that are not explicitly incorporated during the training process. In this case, the generalization is crucial because training may not be completed before a new network event occurs. The use of efficient ML methods like regularization or dropout [33], [230] can be one of the strategies to enhance the generalization of NDTs. However, these methods may affect the model's performance or need to add bias which is also learnable. If the NDT model works well on the training dataset but not so well on holdout sets or faces failure, the generalized model faces the issue of *overfitting*. Improving generalization by reducing overfitting is very important in NDT model training, as it may affect the model's performance. The combination of Reinforcement Learning (RL) and generalization [230] takes care of this condition during the tedious training phase, which can handle incrementally acquired data. The said combination trains the NDT model to *exploit* (use past experiences) [230] and also to *explore* (able to make decisions with new experiences) [230] for better decision-making in an uncertain environment.

IV. RELATED SURVEYS

As DT and NDT have gained recent attention, many researchers have conducted surveys exploring different aspects of these topics (Table 1). A focus on DTs in the context of the IoT was done in [159] to discuss the DT concept with architectural building blocks and key enablers. Based on the assumption that DT is an example of an artifact of softwarization, the authors [159] examined a range of potential development paths for the DT, taking into account its potential application as a key facilitator for the softwarization process. In [234], the recent progress in DTs and the main industrial DT applications were reviewed. Such progress includes DT modeling, simulation, data fusion, interaction and cooperation for DTs, and service. Data fusion is required because DTs must leverage a huge amount of data accumulated from a diversity of spaces (e.g., real environment, virtual space, historical database). With regards to the service notion, a DT can support various services like structure monitoring, lifetime forecasting, and timely maintenance. In [193], the role of AI-ML and big data in building DTs for industrial applications was discussed. Moreover, development tools that assist at different levels of the DT were classified. In [164], the investigation of DT applications within the context of smart cities was made with a focus on basic areas, such as environmental monitoring, urban planning, and smart mobility. Different objectives were followed in [274], and the first focus was on key technologies and technical challenges in NDT, then discussing the well-known smart cities application scenarios such as manufacturing, intelligent transportation systems, healthcare, aviation, 6G networks, and urban intelligence.

Since NDT can play a crucial part in 6G by simulating its functionality to facilitate the experimental analysis easier, the function of DTs in enabling the 6G wireless system was presented in [104]. The authors discussed architectural developments for twin-based wireless systems to improve functional cloning and devise a taxonomy for both different viewpoints (‘‘twins for wireless’’ and ‘‘wireless for twins’’). For the ‘‘twins for wireless’’ viewpoint, they considered issues, such as twin object design, prototyping, deployment trends, physical device design, interface design, incentive mechanism, twins isolation, and decoupling. Also, they considered issues for ‘‘wireless for twins’’, such as access design issues for twin objects, privacy and security, and the design of the wireless interface.

As security is among the most important elements of DT, several studies have also been conducted. In [9], the potential threats associated with DT were classified by examining its functionality layers and the demands on an operational level. The authors provided some security suggestions and solutions that can assist in guaranteeing the suitable and reliable use of a DT. The majority of traditional network modeling tools aim to construct precise data-driven network models that can function in real-time. In [11], it was argued that cutting-edge ML technologies facilitate the construction of some of the NDT's basic components. Moreover, the authors presented a case study that exploits an ML-based NDT to evaluate network performance. They applied this NDT to routing optimization in a use case that is aware of QoS.

The work in [268] focused on the modeling of Complex Networked Systems (CNS) to create an NDT. In [168], a highly interactive 5G emulator was considered as a DT and discussed how DT could be an efficient tool to accomplish the potential of 5G networks and beyond. In [81], DT software technologies, standardization efforts, and various recent projects on DT were surveyed. In addition, the authors introduced a viewpoint of the HyPer-5G digital twin project by considering architectural dimensions. This project is focused on enabling holistic management of 5G IoT networks and beyond.

In [231], a general idea of the Digital Twin Edge Network (DITEN) for 6G was provided. The DITEN paradigm [142] combines mobile/multi-access edge computing and DT. Therefore, it improves network performance parameters and decreases the cost of computation, communication, and caching. In their survey, the authors presented the basic features of DITEN, including concept, framework, and potential. Moreover, they devised a complete design of DITEN that incorporates DT modeling/updating, DT implementation issues, technical challenges, and enabling technologies. Moreover, they provided the characteristic applications of DITEN towards 6G. Example applications of DITEN are wireless systems, healthcare, the IoT, and vehicular networks. For each application, the authors presented the design of DITEN, including DT modeling, DT association, incentive mechanisms, and so on.

As smart contracts are one of the major leverages in blockchain technology, they can facilitate the interactions among DT-based processes, leading these processes to be safely coordinated [208]. Blockchain technology can improve the concept of DTs by guaranteeing decentralized data storage, transparency, data immutability, and peer-to-peer communication in industrial applications. In [287], many advantages of utilizing blockchain in DTs were discussed. Moreover, the authors taxonomized the DT landscape based on basic considerations (e.g., DT levels, industrial use cases, key purposes, enabling technologies, and basic applications).

In [250], a systematic review of the NDT literature was presented and 138 primary NDT studies were analyzed. In [218], key enabling technologies were presented for realizing the capabilities of AI-enabled NDT in 6G, while [151] reviewed the current state-of-the-art DT-enabled 6G-oriented network services. In [237], emerging prerequisites for wireless NDTs were discussed considering the complicated network architecture, tremendous network scale, extensive coverage, and diversified application scenarios in the 6G era. In [256], ongoing DT research was summarized in optical communications and networking. In [206], the privacy and security issues arising from the NDTs were presented. In [40], the universal framework and essential functions of the human digital twin (HDT) was analyzed. Finally, [147] surveys the state-of-the-art research in the area of DTs for natural environments from a networking perspective.

To the best of our knowledge, there is no comprehensive survey that considers how NDTs can facilitate advances in telecommunications and one of its main pillars, network traffic engineering, across its wide range of domains.

All the works could cover a broad range of NDT applications across multiple fields. However, there is a need for a survey of telecommunications and one of its main pillars, traffic engineering, as it gains high priority due to the skyrocketing demand for high data rates. In this way, as the new areas of telecommunications, such as 6G, are expanding and new outing approaches are required for novel use cases, the necessity for emulating and mimicking them with higher accuracy in order to assess behavior and performance improvement seems inevitable, thus, this survey attempts to fill this gap. To do so, it covers a wide range of telecommunication fields and the use of NDT, as well as looking at traffic engineering as an important part of telecommunications and its combination with NDT.

The first step in achieving the goal was to cover the key technologies of NDT for telecommunications. As a result, several technologies that can play prominent roles in establishing NDT for different domains of telecommunications could be covered. This section aims to provide

TABLE 1. Related surveys.

Ref.	Context	Summary
[159]	IoT	It discusses the concept of DT with its architectural building blocks and key enablers.
[234]	Industrial applications of DTs	It reviews main industrial DT applications, such as product design production, prognostics, and health management.
[193]	Industrial applications of DTs	It focuses on the role of ML and Big Data in the building of DTs.
[164]	Industry 4.0, smart cities, smart manufacturing	It reviews the application of the DT concept within the smart city environment.
[274]	Typical application scenarios of NDTs	It presents basic NDT technologies and their technical problems in NDT.
[104]	Wireless systems	It presents key concepts such as design issues, high-level architecture, and frameworks for the DT of wireless systems.
[9]	Security in the use of DTs	It discusses the possible threats associated with the DT concept. It considers its functionality layers and the operational prerequisites.
[11]	ML-based Data-driven Network Models	It presents the NDT concept as a classical network modeling tool. It argues that ML technologies can facilitate the construction of some of its core building blocks.
[268]	Complex Networked Systems	It surveys the modeling of entities and their communications in CNS across regulatory boundaries as they move closer to the crucial objective of developing a DT that perfectly reflects reality. It also evaluates how far the current approaches are from the idealized DTs. It proposes potential directions for constructing a DT that addresses CNS requirements and relies on the integration of CNS and DT.
[168]	5G	It considers DT as a highly interactive 5G emulator. It presents how DT could be an efficient tool to accomplish the potential of 5G networks and ahead of.
[81]	5G	It presents DT software technologies, standardization efforts, and current projects on DT. It presents varied use cases that can be supported by the DT technology. It also discusses the main problems and developments in the research field.
[231]	DT-edge network for 6G	It surveys the DITEN network for 6G. It presents its features and a complete design of this network.
[287]	Blockchain	It examines how blockchain can reform and transform DTs to lead to secure manufacturing that ensures safety, quality, traceability, compliance, and authenticity.
[250]	General context of NDTs	It concludes that the vast majority of the studies on NDTs propose solutions to optimize network performance and aim for security and functional suitability. It outlines that the three most recurrent application domains are those of smart industry, edge computing, and vehicular.
[218]	AI-enabled NDT in 6G	It discusses many applications of AI-enabled NDT in 6G for practical relevance in various industries such as smart cities, transportation, and healthcare.
[151]	DT-enabled 6G oriented network services	It overviews the 6G network requirements for the deployment of DT-enabled 6G oriented network services.
[237]	Wireless NDTs	It explores applications of generative AI, such as the Transformer and diffusion model, to empower the 6G digital twin from multiple perspectives including physical-digital modeling, synchronization, and slicing capability.
[256]	Optical communication networks	It illustrates the evolution of the main parts (i.e., real-time monitoring, mirror modeling, and automatic control) of the DT of an optical network.
[206]	Privacy and security in NDT applications	It provides different privacy and security countermeasures to address the privacy and security issues in NDTs and lists some tools to mitigate the issues.
[40]	Human digital twin (HDT)	It provides an overview of the networking architecture of HDT, including the data acquisition layer, data communication layer, computation layer, data management layer, data analysis, and decision-making layer.
[147]	DTs for natural environments	It surveys the development and implementation of DTs for natural environments from a networking perspective.
Our review	5G/6G, IIoT, smart manufacturing, ITS, smart cities	It explores the applicability of the NDT technology in the context of different networks from a traffic engineering perspective.

the preliminary steps that can help to establish NDT for various areas of telecommunications and also for traffic engineering. After introducing the key technologies, the principal challenges have been presented to shed light on the path of establishing NDT for telecommunications in order to overcome the difficulties.

Subsequently, a comparative analysis of NDT and simulation tools was deemed essential to enable researchers to select the most appropriate tool based on specific research requirements. Then, after the analysis of the necessary pillars, the survey on the use of NDT over networking, telecommunications, and traffic engineering were presented. The collaboration of DT and ML methods with NDT in various telecommunications areas, such as anomaly handling, system state monitoring, resource allocation, and task offloading, was also brought up. Finally, open research challenges related to various parts such as security, privacy, scalability, and cost have been provided.

V. REVIEW METHODOLOGY

Four Research Questions (RQs) that guided our research work were established at the beginning of our study:

1) RQ1: What studies of NDT involved in telecommunications and traffic engineering have been published in the literature?

2) RQ2: What are the studies of NDT, in which NDT is an enabler for improving telecommunications and network traffic engineering?

3) RQ3: What are the various networks described in scientific literature and how NDT can be an assistance to improve telecommunications and traffic engineering?

4) RQ4: What open issues or challenges still require attention?

We created some different Google Scholar search terms. The first mandatory search terms were ''network digital twin'', ''digital twin network'', or ''digital network twin''. The other combined search terms were: ''data models'', ''model training'', ''network simulation'', ''traffic engineering'', ''network management'', ''network optimization'', ''network planning'', ''edge-computing'', ''machine learning'', ''deep learning'', ''federated learning'', ''IoT, ''Industrial IoT'', ''6G'', ''blockchain'', ''vehicular networks'', and ''smart cities''. As recommended by [270], we selected Google Scholar to prevent bias in favor of any one scientific publisher. A time range was not specified by us. Since the search was conducted in the second quarter of 2024, this period and the higher date range limit match. We filtered out duplicates, extensions of previously published research, short papers, and non-English-language publications from the findings, but we kept all documents that presented NDT applications (technical reports, white papers, web articles, etc.). Using the reference list of each paper, we were able to find potential new papers to add to the study by performing a process known as snowballing on the set of found papers. Once more, we used the previously stated exclusion and inclusion criteria. This mechanism helped us to cover a vast range of publications, including many aspects of NDT. Fig. 4 depicts the methodology used to collect and identify the papers included in this review.

VI. KEY TECHNOLOGIES IN NDT

Table 2 shows the main technologies used in NDTs. Hereafter, we briefly describe the key technologies to implement an NDT [274].

A. COMMUNICATIONS

Three types of communication occur in an NDT:

1) PHYSICAL TO VIRTUAL (P2V)

Through the use of wireless communication technology, a physical object can transmit information to its virtual twin, sharing its data in real-time and receiving a response from the virtual twin. Accurate mapping and real-time response/feedback are required to attain viable P2V communications [274]. To achieve this goal, four primary conditions must be met:

- Low communication latency.
- High transmission reliability.
- Data transfer between physical and virtual environments needs to be secure and private.
- The demand from these networked objects must cause the capacity and bandwidth of the network to grow more quickly.

2) PHYSICAL TO PHYSICAL (P2P)

The physical objects are wireless/wired devices such as sensors, actuators, Radio Frequency Identification (RFID) devices, and controllers that can connect with Wi-Fi Access Points, base stations (BSs), and IoT gateways. These objects interact and share information through P2P communications. Communication protocols such as Wireless Personal Area Networks (WPAN), ZigBee, and Low Power Wide Area Networks (LPWAN) technologies (e.g., LoRa and Narrowband IoT (NB-IoT) mainly activate the network connection between them.

3) VIRTUAL TO VIRTUAL (V2V)

V2V communications reflect the communication behavior happening in real space. In the context of Internet of Vehicles (IoV), V2V communications refer to data transmission between the vehicles' entities of the DT model. The computing capability of DT servers (to represent the data transmission behavior) specifies the efficiency of V2V communications. Meanwhile, P2P communications, between physical objects (e.g., real vehicles), mainly depend on the consumption of wireless spectrum resources and radio power. The main contribution of V2V communications is that such communications impose data transmission modeling that overcomes time restrictions in the physical space. While P2P communications between real vehicles require a specific period of time, V2V communications can be implemented very rapidly. Consequently, the communication behavior of physical objects (i.e., vehicles) for a long time can be simulated and mirrored with low time cost. Moreover, a particular communication behavior can take place earlier in the virtual space than in the physical world. This benefit of V2V communications can lead to the scheduling of real IoV resources.

The communication modeling approach is assigned through AI which learns the attributes of the IoV network [105].

B. PHYSICAL DATA PROCESSING

We can implement the DT concept in telecommunications by identifying a group of DT-characterizing features. The authors in [159] defined such features as the 'linkage' between logical and real objects (e.g. IoT devices), the simulation capability, the history of the object's behavior, and accountability/manageability. In general, a DT must satisfy some properties to sustain data processing capabilities. Among them, the most important properties are Representativeness and contextualization, Reflection, Composability, Entanglement, Memorization, and Predictability [159]:

Physical objects often generate massive raw data that could be multi-source, multi-scale, and high noise. As an NDT expands, the amount of data sensed and gathered by real object sensors increases dramatically. Moreover, the transmission of this data through the communication platform could cause congestion. As a result, data must be cleaned to cope with data redundancy and data errors in NDT. Also, data fusion technologies (data generation, modeling, cleaning,

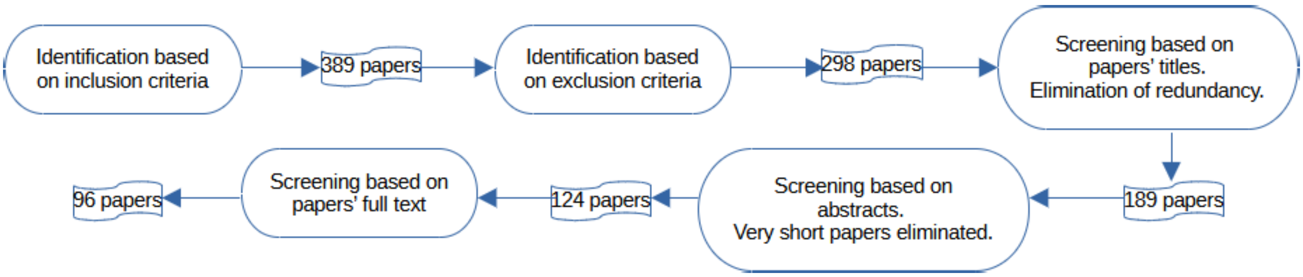

FIGURE 4. Papers selection process.

clustering, mining, and evolution) are required to extract information from raw data and improve data output quality. Data must be fused to enhance the reliability and robustness of the twin data and develop the modeling dimension of virtual twins.

The authors in [274] predicted that sensor data may surpass 100 Gbit/s. So, they described three types of data fusion: Data Fusion for Reduction of Dimensionality, Data Fusion for Matching, and Data Fusion for Expansion. In [197], an order-reduction strategy for DT was proposed which has been used in the high-fidelity generalized method of cells to improve data processing efficiency. Finally, a categorization of data fusion methods was introduced to evaluate various smart city applications in [117].

C. DIGITAL TWIN MODELING

Many researchers suggested DT modeling frameworks to minimize the modeling complexity and make DT modeling feasible to maintain. Also, many studies [130], [211], [234] have been done to design a generalized framework based on an analytical review of research activities and industrial needs. Even though various frameworks have been introduced by researchers, none have considered the characteristics of V2V and P2P, which throws challenges to the collaborative transformation of NTD on a large scale.

Current literature on DT modeling explores three key aspects: 1) specialized models ([158]) tailored for specific application domains, developed using specific modeling approaches; 2) multidimensional models ([163]) that serve diverse functions; and 3) generic models (FlexSim [171], DELMIA [49], Modelica [15], Automad [20], [74], [200], [210]) generated through standard methods. These approaches specifically aim to improve the efficiency and versatility of DT models in real-world applications. DT can give new directions for not yet-known solutions by embedding its functionalities with specific new computing paradigms.

D. CLOUD COMPUTING

Cloud computing allows the dynamic sharing of computing, storage, and other critical resources for large-scale computations using the Internet to fulfill users' ad hoc requests [99]. Big data may be easily integrated with cloud computing by deploying it on cloud servers. Moreover, NDT is focused on the manifested high computational power and centralized processing capabilities that it receives from the cloud. It is technically feasible to analyze enormous volumes of data quickly by implementing NDT on cloud servers. This offers a strong and effective NDT service.

In cloud-based architectures, several storage devices cooperate to support effective big data storage and business access for organizations [185], [274]. As various real-time applications have different requirements, the DTs that were first created with these apps need to be flexible enough to adjust as needed. In particular, the intended NDTs in numerous application scenarios have different needs for latency and computing power, depending on requirements that vary. For example, a real-time road network application would need very fast computing capabilities because this scenario is very dynamic due to the high speed of a vehicle and complex road conditions. The situation is made even more difficult to evaluate because there is a huge number (varying in numbers) of connected vehicles on the road. If this kind of dynamically generated data is not captured as soon as possible, it is likely, that the continuously generated data will overwrite the previously usable data. This may lead to information loss and ultimately it degrades the entire service to a greater extent. Therefore, to handle, manage, and process such massive amounts of data, processing capabilities must be extremely quick to rapidly gather meaningful data.

E. FOG AND EDGE COMPUTING

IoT devices are mostly found at the network's edge. The massive amounts of data that these IoT devices transmit must first be received before being processed. An enormous amount of data must be moved to and from the cloud or data centers via a limited network environment for it to function [183]. However, continuously moving data from IoT devices to cloud servers may cause substantial problem for many time-sensitive applications that require quick responses after swift processing [289]. Overall, this complete system might not be suitable for such applications due to available network bandwidth constraints. Moreover, applications such as location-aware and critical time-sensitive applications might not be able to offer superior performance in conventional distant cloud-based computing systems.

Currently, data centers and the cloud cannot ensure optimal transfer rates and acceptable response times, which are crucial for a variety of ultra-low latency real-time applications. In the end, specialized computing paradigms that occur closer to the linked IoT devices are needed to handle the requirements imposed by bandwidth-hungry, ultra-low latency-sensitive, geographically distributed, and privacy-sensitive applications [99]. Consequently, a more recent computing paradigm, known as ''Fog'' comes into play to effectively solve these problems [25]. The Fog computing paradigm facilitates networking, massive data management, efficient computing, and faster processing, on specialized networking nodes closer to IoT devices. As a result, it efficiently aids in the gap that needs to be bridged between IoT devices and data centers, or simply the cloud [289]. Corresponding to this, industry and academics have introduced ''Edge Computing'' which addresses the aforementioned problems. Fog computing and edge computing have become practical computing paradigms for developing these kinds of applications and systems. By bringing computing resources closer to the IoT device/plane, these paradigms enable localized primary computation [98]. In conceptual terms, all of the processing and analysis tasks can be done in close proximity to the IoT device (or data source), which would maximize speed, network capacity, automation, and maintenance. Edge technology is the entire concept of moving processing, computing, and other associated responsibilities either fully or partially to edge devices.

The goal of integrating edge computing with DT is to experience faster processing and computing capabilities, informal accessibility, better throughput, and low latency performance. In this case, the two technologies complement each other. The DT framework can support both dynamic and tangible static physical objects, which can help the system as a whole integrate with edge technology to satisfy the desired scenarios. The management of various crucial issues, including privacy and security concerns, power consumption, cost, latency, throughput, availability of wireless channels, and maintenance, are greatly supported by edge technology incorporated into NDT. It has been acknowledged that edge technology offers the potential for lowering latency, preserving privacy, reducing power consumption and expense, and boosting dependability in a large extent [274].

1) COOPERATIVE EDGE COMPUTING

Edge servers are deployed in a scattered or geographically dispersed manner. So, the concept of moving processing power to the network's edge further raises the problem of extremely restricted storage and computational capabilities in comparison to cloud data centers. Sahni et al. [205] promoted the notion of assigning administration and analytical duties to the deployed edge devices. For facilitating the sharing of all related computing workloads and data among edge devices, they suggested employing a mesh network that was first built and constructed and connects all edge devices and routers inside the edge network. However, [75] pointed out that when dealing with multiple jobs at once, this strategy may result in longer than acceptable Round Trip Time (RTT) in the system. Such RTT could negatively impact user QoS/QoE. As a result, the idea of edge device cooperation constitutes a solution to this problem.

To manage complex applications, extensive data processing, managing, and assessing of real data are required. In addition, each of these running applications requires a significant amount of computational resources. In this situation, edge device cooperation and cooperative communication can enhance user QoS/QoE. Increased latency can be caused by an edge node with a large number of computing tasks and a big task queue. Other nodes with available computational capabilities should distribute the computational workload across the overloaded nodes [274]. Also, in the fog and edge computing paradigm, the fog/edge nodes receive the sensed data on the IoT/device plane via IoT and end-user devices. In the fog/edge domain(s), many of the accessible fog/edge nodes become overloaded as soon as the amount of sensed data increases. As a result, there is an issue with faster response and delivery times because there is a sudden spike in data processing time at fog/edge nodes due to an increase in computational load. It is necessary for the fog/edge nodes to properly coordinate or synchronize among themselves to prevent such problems. To dynamically shift the load from an overloaded fog/edge node to the underloaded nodes, these nodes must have an appropriate task offloading strategy. Subsequently, it results smaller power usage, optimal resource availability balance across fog nodes, and suitable resource utilization in the system [98].

2) COOPERATIVE CLOUD-EDGE-END COMPUTING

Given how well-suited edge and cloud computing are to one another, this collaborative processing approach utilizes these advantages. The fundamental architecture proposed in [264] shows that the concept of Cooperative Cloud-Edge-End Computing (CCEEC) works on decentralized networking planning, including the cloud domain, edge domain, and terminal (end) domain. These domains collaborate through the notions of Cloud-Edge Collaboration (CEC), Edge-Edge Cooperation (EEC), and Edge-End/Terminal Collaboration (ETC). In this case, edge servers expedite the processing of data that needs a quick response. Cloud servers support significant computational capacity and provide a great means of integrating various forms of data. Additionally, problems with data heterogeneity in the cloud can be mitigated by the dynamic interaction between the cloud and the edge nodes. In addition, the cloud assists in storing a portion of the data by keeping it on hand and sharing it with the relevant client over the network as needed when an edge node's storage capacity reduces [111], [274].

VII. TECHNICAL CHALLENGES AND SOLUTIONS

Although NDT offers significant advantages, many technical issues need to be resolved.

TABLE 2. Key technologies in NDT.

Key technology	Application Areas
Communication technologies	• P2P communication technologies like Wireless Personal Area Networks (WPAN), ZigBee, and Low Power Wide Area Networks (LPWAN) technologies (e.g., LoRa and Narrow- band IoT/NB-IoT). • P2V communication technologies: Wide Area Network wireless communication technology such as LoRa and 5G/6G cellular communications. • V2V communication technologies like Edge intelligence, which consists of AI-empowered edge computing servers.
Physical Data Processing technologies	• Filtering, sorting, aggregation, and classification techniques. • Data fusion techniques, etc.
DT Modelling technologies	• Techniques for building generic models. • Multidimensional modeling techniques. • Techniques for building specialized models tailored for specific communication networks.
Data management, data security, privacy, and data quality	• Digital platforms. • Federated Learning can serve as an enabling technology for ML model training at NDT. • Cryptography and blockchain technologies, big data technologies.
Adaptive decision-making and optimization	• Deep Reinforcement Learning (DRL) technology for enabling adaptive decision-making and optimization. • Multi-Agents Systems (MAS) for improving decision-making, operational efficiency, and simulation capabilities. MAS provides DTs with more autonomy, efficiency, and cognitive intelligence.
Cloud computing for large-scale computation	• Data virtualization, storage virtualization, server virtualization, hardware virtualization, and operating system virtualization. • Technologies for implementing Service-Oriented Architectures (SOA).
Fog computing technologies	• Computing technologies (e.g., computation offloading, latency management). • Technologies such as 5G, WiFi, WLAN, ZigBee, Bluetooth, and Network Function Virtualization (NFV). • Storage technologies. • Naming, identification, and resolution. • Resource management, security, and privacy protection.
Edge computing technologies	• Big data mining in edge computing. • Network slicing in edge computing. • Cooperative edge computing. • Cooperative cloud-edge computing. • Federated Learning–ML edge computing.

A. P2P AND P2V COMMUNICATIONS

In the communication networks of P2P and P2V, extensive and data-rigorous communication occurs that requires low latency, high connectivity, and a suitable network structure [274].

Low Latency: Feedback latency, network latency, data processing latency, and sensor latency all contribute to latency. Low network latency and extremely short data processing times are necessary for time-sensitive applications to provide real-time updates between a real object and a virtual twin. We have two strategies to mitigate this issue: distributed ML and edge computing. Distributed ML can increase the speed and adaptability of data processing by distributing computing workloads among multiple nodes for simultaneous computing [153].

Hyper Connectivity: In NDT, the relationships between P2V and P2P are crucial. Software bugs, power interruptions, and continuous release failures are affecting NDT's connectivity. In various contexts, there are different requirements for network connectivity. NDT may achieve real-time data analysis, enhance asset integrity evaluation, boost operational effectiveness, and lower the likelihood of downtime in static contexts, where the need for connectivity is medium (e.g., for long-term maintenance). Conversely, the high mobility of objects in dynamic scenarios will result in limited network connectivity. In such dynamic contexts (e.g., in intelligent transportation systems), timely input and high connectivity must be provided to handle unforeseen circumstances due to mobility [91].

Network structure: Any NDT architecture needs to offer high reliability and high fault tolerance. In addition, virtual twin instructions need to be flexible methods. The behavior and status of the entire network as well as the real space shouldn't be altered when the virtual space gives erroneous instructions. The effect of the information deviation on the real object may be lessened by using ML to predict the virtual twin's instructions and evaluating them by contrasting them with those produced by the real data [228].

B. DIGITAL TWIN MODELING

NDT modeling depends on a variety of technologies that can provide highly accurate virtual representations of real objects. Nevertheless, scenario-specific modeling cannot be adaptively applied to several applications. This could seriously impede the adoption of NDT technology. Furthermore, the issues with error models and high-precision modeling will determine NDT's reliability [190], [268].

Standardization Framework: None of the existing modeling frameworks can design a general NDT and satisfy the many criteria of virtual modeling. Different domain models need to be standardized to provide a more complicated and comprehensive NDT with a wider variety of applications. To achieve the efficient implementation of many scenarios, the framework must support integrated models with flexibility. By creating the NDT standard framework, the problem of different models' non-interoperability may also be resolved [81].

High-precision Modeling: Until now, there is still little research about multilevel and multi-dimensional advanced modeling technologies. Conventional modeling (e.g., simulation languages) has drawbacks like poor flexibility, complex configuration, and error-proneness, and it is too simplistic to obtain the comprehensiveness and accuracy required by NDT. To build a trustworthy NDT, modeling and simulation technologies require further advancement [274].

The Model Keeps Updating: High-fidelity models of physical objects are not available, and most physical objects have unclear underlying principles. Hence, it is extremely difficult to diagnose and predict using partial knowledge and ambiguous principles and to continuously update models based on principles and data. Another difficulty with DT modeling is that a model needs accurate data and sufficient processing power to support its ongoing updates [190], [244].

C. COMPUTING

Apart from the advantages of combining NDT with cloud and edge computing, the following issues must be resolved [274].

Architecture: NDT implementation requires a lot of processing and caching power. While edge and cloud computing may be able to satisfy these needs, a standard architecture that facilitates edge-enabled NDT operation is still missing. NDT's connectivity, data properties, resource limitations, management effectiveness, and the coordination of applications, services, and resources must all be considered in such architecture.

Resource Limitation: Edge nodes have a limited amount of computing and storage resources. An intelligent algorithm must be used to analyze the data from the virtual model. Furthermore, the use of intelligent algorithms necessitates storage resource support. It is necessary to examine how to store various models for data analysis at the edge nodes to meet the demands of various real network objects. Enhancing data analysis efficiency is vital to guarantee precise feedback to physical systems. The data analysis process will require a significant amount of processing power, but edge nodes have limited processing power. Reducing the strain on edge nodes (through a migration task strategy) can enhance the feedback of data quality to real objects.

Environmental Dynamics: The dynamics of task requirements and NDT connections are the two main environmental dynamic challenges. In edge computing, real objects are mainly linked to edge nodes via wireless networks. However, wireless channels have reduced anti-interference capabilities and are prone to signal interruption, particularly in environments where objects move at high speeds. Physical object movement will cause the communication system to become unstable by upsetting the initial state of resource allocation for communication. Numerous physical object types exist for the dynamics of task requirements in NDT, and the corresponding computing task requirements will also vary greatly. The data produced by physically similar objects may differ, and the needed resources are diverse. Optimizing the distribution of scarce resources at the edge is important to meet the demands of NDT's dynamic computing tasks.

To facilitate Cloud-Edge-End computing, Wang et al. [264] introduced a collaborative framework aided by DTs and designed for carrying out task hierarchical offloading. Their framework is composed of two layering architectures to provide flexible offloading modes and lessen the computational load on edge servers. Their technique presented the idea of implementing a physical layer network infrastructure consisting of edge servers, user-ends, and cloud servers. Also, their technique presented the Digital Twin Layer which contains various DTs. The use of these DTs is driven by the expectation that they will dynamically correlate with their physical counterparts, facilitating quick and seamless data sharing and decision-making. The proposed DT-assisted scheme addresses critical parameters such as offloading modes, workload split ratio, bandwidth allocation, and optimized device associations to reduce system costs, concerning power consumption and workload latency. Further, the authors implemented a MADDPG to effectively accomplish Cooperative Computation and Resource Allocation (CCRA). They validated the provided algorithm's performance in terms of task success rates, power consumption, and latency through extensive simulations over an IIoT scenario. However, in the context of IIoT, integrating ML-based dynamic policies with edge devices will be a very complex task due to limited resource availability at the edge devices. In addition, severe problems related to effective communication with the distant cloud (due to increased latency and raw data privacy) contribute to this complexity [157].

Khan et al. [104] discussed two design aspects concerning dynamic wireless systems and DT: (i). Twins for Wireless (TfW) and (ii). Wireless for Twins (WfT). The Twin design often addressed by the TfW design component mainly talks about the effective implementation of services for enabling optimum networking functionalities for wireless systems by employing the notion of DT. On the other hand, the WfT design element addresses efficient communication modeling to put signaling protocols for the implementation of twins into practice. The optimization of resources for wireless systems, to facilitate twinning across wireless networks, is addressed by the WfT design component. Wireless resources can be employed for enabling twinning through two operations: (i) twin physical objects/systems training and (ii) twin operation signaling. It is evident that data or learning updates will be transferred to train the twin, and that network resources will be employed more effectively to enable that transferring function. End devices must send their data to a common central storage (which may be located on an edge server or in the cloud) if single-point-based centralized learning methods are implemented. But now, there are a lot of smart applications that constantly generate a tonne of data. It will take a lot of communication resources to move this enormous volume of data to a central storage location for further processing, which may not be available at that specific time and add to the application's unacceptable latency. In the interim, this entire raw data transfer process can lead to another privacy breach issue. Therefore, a distributed learning mechanism, called Federated Learning (FL), is needed to handle the above challenges.

D. FEDERATED LEARNING

Today, there is a prevailing idea of accomplishing comprehensive ML training at the network edge (usually over wireless communication settings) while taking consistency, accuracy, and latency into account. Considering this perspective, it is essential to design independent edge-computing-based ML architectures. Such architectures must support optimized effective operations capable of adapting to communication expenses, wireless channel dynamics, and associated events. Moreover, these operations must be able to offer support by effectively coping against straggling devices. As some sophisticated devices may be slower or less dependable throughout the training phase, such an architecture must handle these differences to maintain the overall accuracy and efficiency of the training process. Furthermore, un-modeled phenomena (that might not be included in the training data) should be managed by the architecture and its operating methods. Along with the previously mentioned challenging factors, a well-designed ML architecture also needs to take into account the restricted resources that are available at the edge device end. We must consider limitations on power consumption, storage capacity, and computational resources. McMahan et al. [152] and Konečnỳ et al. [110] investigated a distributed learning scheme (i.e., FL) that does not require central data storage and allows users to leverage shared models trained from huge data. FL is based on the idea of transmitting only the learning model update rather than the whole data. This idea results in decreased communication resource utilization [129].

For FL-based ML edge-computing architectures, edge devices involved in the training process periodically exchange neural network parameters (usually called weights) and gradients (the modifications needed to update these shared weighted parameters). Such parameters are estimated and approximated during local training. However, there are many participating devices, and their Internet connection is either sluggish or inconsistent. Since many edge devices are collaborating, managing highly unbalanced and non-Independent, and Identically Distributed (NIID) is quite challenging and poses numerous issues to practically implementing FL. In such an environment, local updating and low collaboration and cooperation from edge devices pose several other challenges in the system. Considering the poor and unstable Internet connection problem of edge devices, the authors [153] presented two methods to reduce data transmission from edge devices to the centralized server during training:

- The *Structured Updates* technique which learns updates using fewer variables in an attempt to decrease the volume of data.
- The *Sketched Updates* technique which encompasses transmitting an entire update in a compressed form.

Both techniques and other associated mechanisms for federated settings are still in their early stages of development and need more clarification for their successful implementation. Also, they need to address numerous other challenges, including the ML-communication effective [176] design, while accounting for dynamic wireless channel properties and events, and edge device resource constraints. Here, the typical FL problem comprises learning and developing a single global statistical model utilizing data distributed across millions of small-scale or large-scale edge heterogeneous devices. The learning process has been carried out under the strict restriction that any generated data will only be processed and kept locally at the edge device level, with only real-time updates being communicated with the centralized server periodically. Based on such settings and constraints, Li et al. [123] solved the distributed optimization problem by minimizing an appropriate objective function. However, they outlined a few obstacles that exist in the federated environment:

1) **High Communication Cost:** Wireless connectivity ultimately operates as a bottleneck even though it is an essential backbone for federated environments. Additionally, there are privacy issues associated with the sharing of raw data as well. Therefore, it is crucial to store the data locally on each device instead of transferring it to a single location [123].
2) **Heterogeneous Edge Devices:** We have different hardware types and components in terms of CPU and memory, available power level, as well as different network connection types (3G, 4G, 5G, and WiFi). Such heterogeneity can result in a significant variation in the storage, processing, and communication competencies of each heterogeneous edge device in the federated environment [24].
3) **Statistical Heterogeneity:** Over time, edge computing devices gather and produce data in a non-uniformly dispersed way throughout the federated network. The devices generate inconsistent and non-uniform data since mobile device users may use different languages or have different typing practices, which eventually results in non-uniform data. In addition, every device

has the potential to collect and process varying amounts of data. Typically, FL leverages fitting a model to data generated by distributed edge devices. Throughout the federated network, every edge device gathers data in an NIID pattern, with each node having a unique distribution of data generation. The information gathered from various devices may show trends or connections. These patterns might show the relationship between devices and the dispersion of their data [222]. In the case of classical distributed optimization systems, it appears that this data generation process is confined to the simple premise that data will always be in NIID alone. This may lead to an increase in complexity in the areas of problem modeling, theoretical testing, and empirical solution assessment [123].

E. FEDERATED LEARNING WITH DIGITAL TWIN IN CLOUD/EDGE ENVIRONMENTS

The idea of *Mobile Edge Computing* (MEC) was introduced to control and provide a decent QoS for delay-sensitive applications. Nonetheless, in the perspective of Industry 4.0, dynamic real-time collaborations are required to close the distance between physical and digital systems. In general, in real-time communication networks, communication delays are impossible to predict precisely. Further, continually generating and transmitting real-time big data makes it challenging for MEC servers to accomplish online optimization procedures while managing and analyzing the Channel State Information data from IoT devices competently. The DT paradigm can close the distance between cyber and physical systems for improved optimization of the process. Even if DTs are important in this situation, comprehensive evidence is still needed in the light of DT modeling and its uses in communication networks, particularly in wireless scenarios [142]. Furthermore, synchronizing vast volumes of data is the biggest issue because processing and communication capacities are constrained in these settings, leading to inefficient DT models, especially in the context of IIoT networks.

Many authors [142], [142], [143] adopted FL in their Digital Twin-assisted models. Lu et al. [142] suggested a method for developing and executing a DT of edge networks, which uses FL for locally retraining aggregated models, particularly on edge devices. Another benefit of using FL is that their strategy can optimize communication procedures and performance by employing efficient bandwidth allocation and user scheduling techniques. Finally, the scheme's novelty is its ability to avoid superfluous raw data transmissions by performing recurrent retraining of aggregated models locally. The authors enhanced their earlier framework [142] in the area of communication security and proposed a new system [142]. They [142] re-introduced the employment of DTs with edge computing technology, focusing on edge networks, and then offered the revolutionary DITEN paradigm. The goal of this paradigm is to close the distance between digital systems and physical edge networks. The new aspect of their proposed system [142] is that it uses blockchain technology to store the edge device's aggregated model critical parameters on base stations, which improves the FL procedure's robustness to data privacy problems. The authors in [143] presented a strategy, called Digital Twin Wireless Networks (DTWN), for minimizing the imprecision of learning parameter transfer due to lengthy communication distances and restricted communication resources (between edge servers and end-users). Their built FL and blockchain mechanism work together in the introduced plan to ensure DTWN's reliability while also protecting users' data privacy. Moreover, considering the challenging and unreliable 6G network environment, Sun et al. [229] developed a mobile offloading policy on top of the DITEN paradigm. Their method is motivated by the fact that the offloading strategies currently in use that are based on Mobile Edge Computing (MEC) technologies completely overlook the effects of user mobility and the dynamic nature of 6G MEC networks. This proposed scheme makes use of and considers the DITEN paradigm followed scenario, wherein DTs help make critical offloading (mobile) decisions by providing DRL agent training data and approximating the current states or the availability of computational resources at the edge servers. Therefore, by taking into account the very constrained migration cost that usually arises in the situation of user mobility, this strategy (based on the aforementioned scenario) contributes to minimize the offloading latency. The authors in [43], [104] highlighted the requirement of substantial resources to train DT models across wireless networks utilizing distributed learning. They also highlighted how wireless channels' intrinsic traditional parameters lead to a deprivation in communication quality, which in turn results in inferior performance of distributed learning-based twin models. This implies that when learning model updates are being transferred between end devices and the centralized server, the performance of such models will be negatively impacted by the wireless channel. The authors [48] suggested an optimal policy for resource allocation and offloading computations by contemplating the stochastic task arrivals. They initially built an NDT model that aims to establish an effective mapping between digital and IIoT systems. Concerning an IIoT network with heterogeneous resources, they built an NDT model and addressed the problem of how difficult it is to simultaneously reduce energy consumption, make the best decisions for task offloading, maximize computational and bandwidth efficiency, and handle scenarios involving time-varying wireless channels. Though it is still in its infancy, all these researches present a new paradigm that uses edge computing and NDT.

With the widespread deployment of 5G mobile networks and their commercial availability, academics and industry professionals wait the arrival of 6G mobile networks. Novel technologies such as the utilization of higher frequency bands, smart surfaces, and Orbital Angular Momentum multiplexing, and some networking possibilities, i.e., massive

Machine Type Communications (mMTC), ultra-Reliable and Low-Latency Communications (uRLLC), enhanced Mobile and BroadBand (eMBB) have been presented for this purpose. Additionally, realizing physical, network as well as Medium Access Control (MAC) layers via molecular, visible light, and quantum communication, and the use of AI and ML procedures are still in the research and development stage. Several of these innovative technologies and paradigms might be seen as 6G enablers [1]. Here, eMBB manages data transfer between numerous networked elements, including end-user smart devices, edge devices, and cloud servers. On the other hand, mMTC focuses on administering many connected and complex devices like wearables, actuators, and sensors, through dense urban deployment. Lastly, uRLLC is responsible for overseeing highly time-sensitive communications, such as those involving vehicular communication, base stations, and edge devices [99], [284]. These cutting-edge 6G technologies and paradigms offer the possibility of creating a contextin which DTs of a network (or object) can be used to create and preserve a widely used virtual representation of a real network. Although numerous DT-based technologies have already been developed to support industrial applications advocated by 5G or even 4G, such a system is not being utilized and thoroughly accepted in other sectors so far. Furthermore, the associated bottlenecks to realizing the potential of DT include widespread communication, improved throughput performance with minimal battery consumption, and increased reliability. These necessitate the use of beyond-5G networking technologies, hence, advocating the use of 6G as an enabler for successful adoption of DTs [1].

In reality, all applications have to cope with the well-known problem of power, memory, and storage being highly scarce in today's networking environment. Thus, substantial resource management is needed to competently sustain DT technology, including data generation, data communication, data analysis, and complex computations. The idea that more data should be handled and stored locally at the edges of the network is widely accepted. And this strategy won't change for 6G and subsequent futuristic networks [1]. Further, it will be necessary to host different back-end systems and solutions across several data centers (both in the cloud and at the edges of the network) to support DTs properly. DTs are guaranteed to function effectively and manage the required data processing and storage thanks to this dispersed approach. In addition to these difficulties, the authors in [1] emphasized replicating DTs because there is a chance that the edge or cloud servers' processing and storage capacities could result in a number of system-oriented limitations that could make it difficult to implement DT in a single location, resulting in performance bottlenecks. Moreover, the authors in [1] highlighted that network links' and servers' failure hinder the seamless communication between a physical twin and its corresponding DT. Consequently, having numerous DT replicas distributed around the cloud and/or the edge servers is crucial. Also, numerous elements of cloud and/or edge-distributed DTs are required to ensure seamless data communication and efficient training of AI-based models to initiate automated and smart procedures for applications, this complete procedure is well-known as *federated DT*.

VIII. NDTS COMPARISON WITH NETWORK SIMULATORS

Network simulators are essential tools in the field of network design and development, offering a cost-efficient means to experiment with network configurations, topology alterations, and protocol adjustments without the financial and operational risks associated with real-world implementations. Consequently, the majority of researchers initially evaluate their novel approaches using network simulators before deploying them in real-world environments. A variety of simulators exist, each designed for specific use cases, including wired, wireless, or hybrid networks. Generally, network simulators are computational models that aim to replicate network behaviors with a high degree of accuracy, encompassing various network layers, from transport and routing protocols to the data link layer. However, these simulators often face significant challenges related to the substantial computational resources required, which escalate with increasing network complexity, size, and the volume of simulated data [66], [177].

There are several simulation tools in the field of networking area, such as NS-3 [198], Objective Modular Network Testbed (OMNeT++) [247], Network Simulator (NetSim) [192], JavaScript SiMulation (JSIM) [223], REalistic And Large (REAL) [103], Optimized Network Engineering Tool (OPNET) [38], QualNet [3], [248], Global Mobile Information System Simulator (GloMoSim) [293], Tiny OS SIMulator (TOSSIM) [119], Dynamic Routing Model simulator (DRMsim) [85], and Traffic and Network Simulation environment (TraNS) [179] that each offers distinct advantages and disadvantages, making it essential to select the most suitable one based on the specific needs, requirements, and scenarios of the simulation. This section provides a brief overview of the most commonly used simulators, discussing their strengths and limitations, and considers how NDT might serve as an alternative to these tools.

A. A CLOSER LOOK AT THE SIMULATION TOOLS

Among the simulators discussed, NS-3, OMNeT++, NetSim, REAL, OPNET, and QualNet are the most widely utilized, encompassing a broad spectrum of network types [177]. NS-3 is a free open-source simulator widely used in the scientific community that supports C++ and optional Python scripting. The modular structure of the tool makes it feasible to extend the simulator and add new features. It also provides some visualization tools, which is another advantage for the simulator; however, it is often considered basic compared to those offered by other simulators. Some other advantages of the simulator are that it has an emulation mode that allows more accurate analysis, the ability to run user space and kernel space protocols relying on Direct Code Execution (DCE), and it has an active community. In conclusion, NS-3 is a widely used open-source discrete-event network simulator designed for research and educational purposes and provides a comprehensive simulation environment for modeling various network protocols and systems. However, it suffers from some drawbacks, the main one of which is reliability. The results obtained from the simulation scenarios cannot guarantee that the implementation of the new approach would yield the same in a real network, making it difficult to rely on the simulator in practical scenarios. Moreover, when there is a large amount of traffic, since it is a packet-level simulator, as the number of packets increases, the simulation time and resource consumption will also rise. Furthermore, it does not have bidirectional communication with the real network to update its data or send control signals to the network [177], [198]. Of these issues, NDT can address most of the issues, as its accuracy is high, the execution time is low, and it has bidirectional communication with the network. In addition, NDT can bring another privilege that the other simulators cannot, which is that a user does not need to spend much time learning how to use it [66].

OMNeT++, like NS-3, is another discrete event simulator, as both attempt to simulate the events of a network as timely sequences. The main goal of OMNeT++ was to design a general-purpose simulator that could cover a wide range of networks. This goal could be accomplished to some extent as it can simulate many networks, such as wireless, queuing, ad-hoc, peer-to-peer, optical switch, and storage area networks [247]. OMNeT++ is also a free, modular, and extensible simulator similar to NS-3. Some other advantages of OMNeT++ are that it also has a graphical environment, supports parallel simulations, and has IPV6 capability. Overall, OMNeT++ is recognized for its precision and depth, making it a valuable tool for those involved in the design and analysis of complex communication networks. However, it cannot support a wide range of protocols, like NS-3, suffers from the reliability issue as the accuracy is not as exact as the real networks, and is resource intensive [177], [247]. In addition, OMNet++ has some documentation, a manual, a developer's guide, and an API reference. In conclusion, OMNeT++ is a discrete-event simulation framework primarily used for modeling and simulating communication networks. It is an open-source, modular, and extensible simulation environment that is widely used in both academia and industry for research and educational purposes. However, more efforts should be put into making it clearer and more up-to-date [2]. Moreover, it is resource-intensive, has some user interface limitations, and has a small community. As the issues are similar to those of NS-3, NDT can solve them mostly by providing an accurate and fast framework [247].

NetSim [249], is another discrete event simulation. Developed by Tetcos, NetSim is known for its user-friendly interface and detailed simulation capabilities, making it a versatile tool for studying a wide range of networking scenarios. This commercial simulator can also cover a wide range of networks, protocols, routers, and more. However, the tool is costly, resource-intensive, has performance degradation as the size and complexity of the network increase, and can take some time to run simulations [177]. Moreover, the commercial license of the tool is a hurdle in society's development of the tool by adding new features.

REAL focuses on packet-switched networks and provides the ability to analyze flows and congestion status in the network. The tool supports the most commonly used flow controllers such as Transmission Control Protocol (TCP) and fair queuing scheduling such as First-Come First-Served (FCFS). The tool was developed primarily for the academic and research communities to model and analyze the behavior of computer networks. Originating from a project at Stanford University in the early 1990s, the REAL Simulator was designed to be simple, flexible, and efficient, making it a valuable tool for studying network protocols, particularly in the context of TCP/IP networks. To achieve its goal, it contains thirty basic modules, which have been written in the C language. One of the main advantages of REAL is its scalability, like that of other modular simulators. However, it is suitable for small-scale simulations; Furthermore, the graphical interface is less responsive compared to more modern simulation tools [103], [177]. Both issues can be addressed by using an NDT, as it can cover a wide range of scenarios due to its data-driven approach. In addition, the response time of an NDT can typically be much faster than that of simulation tools.

OPNET [38] is another simulation tool providing a comprehensive development environment for analyzing and simulating the performance of communication networks. The tool covers a wide range of networks, from simple Local area Networks (LANs) to complicated satellite networks. Like many simulators, it is also a discrete event tool and supports graphical user interfaces. OPNET is also a useful tool for investigating the protocol and application parts of a network relying on its features and is a powerful network simulation and modeling software. The tool was originally developed by OPNET Technologies, Inc., which was later acquired by Riverbed Technology. The software is widely used in academia, industry, and government for designing, analyzing, and optimizing communication networks, including both wired and wireless systems. However, its commercial license makes it difficult for the community to use, develop, and extend the tool by adding new modules. In addition, the simulation process consumes a large amount of memory, is costly, inflexible due to having fixed protocol models, has a small community compared to that of NS-3, and the tutorial manual is not sufficient. Another important aspect of OPNET is that it is mostly aimed at operations rather than research [38], [144], [177]. The memory consumption, hard-to-learn process, buggy characteristics, complex graphical user interface, and all the other problems for OPNET can be addressed by the features that NDT can bring.

QualNet [3], [248], developed and provided by Scalable Network Technologies, Inc., is a sophisticated network simulation tool. It serves as a comprehensive platform for planning, testing, and training, by providing a highly accurate emulation of the behavior of physical communication networks. In addition, QualNet facilitates the evaluation of mobile communication networks with a level of speed and realism. The tool utilizes advanced Parallel Discrete Event Simulation (PDES) algorithms that are specifically engineered to harness the capabilities of multi-core and parallel processors. This approach significantly enhances the event processing rate, thereby accelerating simulation execution speeds. As a result, QualNet is able to perform high-fidelity simulations of large-scale networks at high speeds. QualNet has an optimized memory management method, is capable of simulating large-scale networks, has a user-friendly interface, and has professional technical support. However, it is costly, resource-intensive, not open source, and has a small community [3], [177], [248].

In conclusion, network simulation tools are essential for designing, testing, and analyzing network performance without the need for physical infrastructure. However, these tools have various flaws and limitations that can affect their accuracy and reliability. Additionally, depending on the tool used, they may suffer from other drawbacks, such as simplified models, scalability issues, complexity, the need for specialized knowledge, lack of real-time interaction with the physical network, and high resource utilization. The use of NDT to mimic the network can address these issues to some extent, as it is highly accurate, easier to scale, does not require users to learn complex processes, supports bidirectional communication with real infrastructure, and can emulate complex scenarios. Table 3 compares the main features of NDT and simulation tools.

B. COMPARISON OF NDT AND SIMULATION TOOLS

As network management and design continue to evolve, advanced technologies have introduced powerful tools for analyzing and optimizing network systems. Among these, NDT and simulation tools stand out as essential resources. Each offers distinct capabilities and insights tailored to different aspects of network management. Their comparison is intended to clarify the roles, benefits, and limitations of NDT and simulation tools, and to provide a thorough understanding of how each contributes to effective network management. NDT technology is a state-of-the-art solution that creates a real-time virtual replica of a physical network. This digital twin constantly integrates live network data, facilitating ongoing monitoring, analysis, and management. Simulation tools, on the other hand, are designed to model and analyze hypothetical scenarios and network configurations. They allow network engineers to test different designs, configurations, and conditions in a virtual environment without affecting the real network [11], [66], [177].

The comparison of NDT and simulation tools encompasses several aspects. The first one is about the purpose and application, where NDT offers real-time analysis and enables immediate network management, while simulation tools come up with strategic planning and scenario analysis. NDT is useful when there is a need for real-time bidirectional communication and accuracy, however, simulation tools can act as closed boxes where the analytical results are provided [11], [28]. In this case, it is time-consuming to build an NDT, but it can provide more advantages.

Another important consideration is the creation, data integration, and deployment of these tools. NDT relies on real-time data from the physical network to build its models, thereby closely reflecting the network's actual performance. In contrast, simulation tools use historical data or hypothetical conditions to model different scenarios, facilitating experimentation and prediction. Furthermore, simulation tools are typically discrete event tools that model events as sequential occurrences. Discrete event simulations are indeed a widely used methodology in simulation tools to model events as sequential occurrences. However, they have their own hurdles in emulating the network with high accuracy [161], [173], [221], [243].

Another principal factor is the operation deployment and the extent to which it is realistic to use them for this purpose. NDTs are designed specifically for real-time operational management. They provide a continuous, up-to-date view of the current state of the network by integrating live data from the physical network. This enables the creation of an accurate and dynamic model that reflects the ongoing performance and status of the network. Simulation tools, on the other hand, focus primarily on strategic planning and scenario analysis. They facilitate the exploration and evaluation of different network configurations, designs, and conditions by modeling various "what-if" scenarios, but they do not provide accuracy close to real networks. The reason is that, while simulations can model various scenarios effectively, they may not always capture the full complexity and real-time dynamics of actual networks [11], [161], [173], [177], [274].

Finally, the methodologies and implementations are different in both. NDTs rely on continuous streams of data from multiple network sources, such as sensors, network management systems, and performance monitoring tools. These streams provide real-time insight into network performance, traffic patterns, and overall system conditions. In addition, there is communication between the twins to update their status to more closely resemble the actual state of the network. On the implementation side, data integration for NDTs involves aggregating and processing real-time information through Application Programming Interfaces (APIs), data brokers, or direct connections to network devices. This collected data is used to dynamically update and refine the digital twin model. On the other hand, simulation tools develop virtual models of the network using hypothetical scenarios or historical data. These models allow users to experiment with different network configurations,

TABLE 3. NDT and simulation tools comparison.

Metric	NDT	NS-3	OMNeT++	NetSim	REAL	OPNET	QualNet
Real-time synchronization	Reflects real-time changes	does not inherently support real-time synchronization	does not natively support real-time synchronization	offers real-time synchronization capabilities to some extent but with some limitations	Provides realistic simulation environments including wireless and mobile networks	Does not inherently provide real-time synchronization	Support real-time synchronization in certain contexts
Scenario modeling	Simulates various hypothetical situations and provides robust scenario modeling capabilities	highly capable for scenario modeling, especially for research purposes	cannot support a wide range of protocols, like NS-3	Provides robust scenario modeling and Supports a wide range of networking scenarios	Offers robust scenario modeling supports	Provides robust scenario modeling capabilities	Is highly regarded for its scenario modeling capabilities, offering robust, flexible, and detailed modeling features
Predictive analysis	Providing future predictions	Needs analyzing the results to forecast system behavior	Is not explicitly designed for predictive analytics	does not support built-in, advanced predictive analysis capabilities	Offers some predictive analysis capabilities, but its focus is more on real-time simulations	Highly capable in the realm of predictive analysis	Offers powerful predictive analysis capabilities
Visualization	Provides visual details of network components and behavior	Provides some basic visualization tools	Has a graphical environment but has some user interface limitations	Has robust visualization capabilities and a user-friendly interface	Offers powerful visualization capabilities but the graphical interface is less responsive	offers advanced visualization capabilities and supports graphical user interfaces	Is designed to help visualize network topology, traffic flow, protocol interactions, and performance metrics
Integration with physical networks	Integrate seamlessly with physical network devices	can be achieved through emulation capabilities	Supports integration with physical networks to some extent	is not specifically intended for direct integration with physical networks	Supports integration with physical networks	Has limited direct integration with physical networks	Offers integration with physical networks
Speed	Fast ad efficient enough if suitable models are used	Is known for its high efficiency and fast simulation performance	When there is a large amount of traffic, speed can be impaired and is normally slower than Ns-3	NS-3 is generally faster and more efficient than NetSim	Generally, slower than NS-3 and the speed depends on various factors, such as the complexity of the network being simulated	The simulation process consumes a large amount of memory and is costly	Depends on the complexity of the scenario and is generally slower than Ns-3
Scalable	Yes	Yes, but it depends on the complexity of the simulated model	Is resource intensive	Scalable but it is resource-intensive, has performance degradation as the size and complexity of the network increase, and can take some time to run simulations	Yes, but it depends on the complexity of the simulated model	is highly scalable but the tutorial manual is not sufficient	Yes but its scalability largely depends on the size of the network
Model accuracy	Accurate due to data-driven models	Highly accurate, however, the accuracy depends on several factors, and is can be less accurate than NDT	Accurate, however, the accuracy largely depends on the quality of the models moreover supports parallel simulations	accurate for simulating standard networking protocols and typical enterprise-level network scenarios	The accuracy can vary depending on the simulation specifics	Is widely recognized for its high degree of accuracy, however, the accuracy depends on various factors, including the network models	High particularly for large-scale, real-world network simulations
Validation (Results can be compared to real-world ones)	Highly reliable	reliable (to some levels)	Highly reliable; however, it depends on the quality of the models	results can generally be compared to real-world scenarios to some extent	Is designed with a strong focus on real-world validation	It depends heavily on the specific use case, and real-world data availability, however, it is generally considered good	Offers robust validation capabilities
Cost	No need to license payments	Open-source	It is free	It is not free	It is not free	It is not free	It is not free
Ease of use	Easy to model	Should be learned	Should be learned, however, one of the easiest-to-use network simulators	Should be learned but is generally considered easy to use	is designed to be user-friendly, but the need to learn the tool is remaining	Should be learned but it is feature-rich, professional-grade simulation tool	Should be learned, however,it is considered user-friendly
AI/ML integration	Capable	Yes, but with the help of external libraries	It doesn't have native AI/ML libraries built-in but supports AI/ML integration	offers AI/ML integration capabilities	Does have capabilities for AI/ML integration	Does not natively provide integration with AI/ML	Does not have native support for AI/ML integration

traffic patterns, and operating conditions. Scenarios are also specified by parameters such as network topology, traffic load, and device settings. Simulation processes run these scenarios to assess how different conditions affect network performance [66], [224], [274].

To sum up, both NDT and simulation tools are essential for developing new ideas and approaches for networking, but they address different needs and use different methodologies. NDTs provide real-time operational control through accurate and dynamic models that continuously integrate live data from the physical network. This capability enables immediate insight and informed decision-making based on real-time network conditions. Moreover, NDT has bi-directional communication with the physical network, is well developed, can achieve high accuracy and speed, and is easy to learn how to use. In contrast, simulation tools are designed for strategic planning, providing a virtual platform to evaluate different network configurations and hypothetical scenarios. While they may lack the real-time accuracy of NDT, simulation tools are critical for forecasting network performance and assessing the impact of potential changes. A comprehensive understanding of the strengths and limitations of both NDT and simulation tools is key to optimizing network management, and should be applied based on needs and requirements.

IX. TELECOMMUNICATIONS AND NDT: SYNERGIES AND APPLICATIONS

NDT has various applications in different domains of telecommunications, from 5G/6G and smart cities to IIoT. Overall, NDTs facilitate enhanced decision-making, efficiency, and innovation by providing a comprehensive and dynamic view of complex systems. As a result, this section aims to discuss the progress in NDT applications across telecommunications, industry, smart cities, and intelligent transportation.

A. NDT AS AN ENABLER FOR 5G/6G NETWORKS

As mobile communications evolve, more novel use cases and features are covered by the new generations. The pace of this evolution has been accelerated during recent years with the deployment and development of new generations of mobile communications, such as 5G and 6G. These deployments can bring many new capabilities, such as new spectrum deployment, redesigning the core network, network slicing, full realization of the IoTs, high data rate, and ultra-low latency [70]. This rapid development will cover more than 22 million jobs by the year 2035 [34], indicating its importance in the future society and the necessity of dedicated focus on it. However, there are many challenges in the way of 5G/6G implementation, such as blockage, misalignment, full realization of network slicing, SDN/NFV deployments, and more [70], [182], [296].

There have been many novel approaches to address these issues, which have been analyzed under different simulation tools or in practice. Testing a new approach in a real 5G/6G environment presents its own challenges due to existing limitations [181]. For the simulation scenarios, there have been some attempts to create suitable ones, such as NS-3 Millimeter wave (mmWave) [156], however, they cannot replicate the network with the same level of precision [71]. These limitations show the requirement for a framework that can be used as a test platform for 5G/6G networks. As a result, NDT has recently started to be used to assist in analyzing various aspects of 5G/6G.

NDT can help to virtualize 5G/6G networks in various aspects. To do so, an NDT architecture for 5G provides an environment to create a simple 5G NDT, which can then be evolved to comprehensively cover the 5G network by using various AI methods. This mechanism not only provides a framework for analyzing the 5G network, but also the precision can be improved during the process by updating the data and incorporating new scenarios and use cases. Replacing 5G/6G simulators with the NDT can bring many benefits, such as improving the execution and validation time as it is faster and more accurate, providing reliable results that can bridge the academic and operational, reducing costs, enabling bidirectional communication with the network, which provides the potential for problem diagnosis, and facilitating network slicing analysis, as an individual twin can be assigned to a particular slice [168], [175].

A guideline on how to use NDT to model the various characteristics of a 5G network, such as communications, devices, links, applications, and operating environment, has been done in [199]. The paper provdes a method that can automate the creation of NDT for a 5G network. The main direction of the research was toward providing a framework for the so-called Mobile Network Digital Twins (MNDT), which aims at the establishment of NDT for mobile networks. The framework automates various phases of the NDT creation, such as data acquisition, modeling, adaptation, deployment, interconnection between the twins, and the feedback phase. One of the most important aspects of the work was the use of some software agents in the physical networks through some protocols, such as Internet Control Message Protocol (ICMP) polls and Simple Network Management Protocol (SNMP) queries, which helped to create the data repository for the network modeling process as an important step towards having MNDT 5G platforms.

In addition, [78] provided a detailed analysis of the requirements for creating NDT for 6G networks. Terahertz communications, SDN, Space-Air-Ground Integrated Network (SAGIN), federated learning, and blockchain were introduced and studied in detail as key enablers for NDT towards 6G. However, the paper also addressed various challenges associated with the implementation of these technologies, highlighting issues that need to be addressed.

Other approaches have emerged that aim to create NDT for 5G/6G, focusing on the security and management dimensions of cellular networks, such as the B5GEMINI project [101]. The first step of this framework was to provide a platform for data collection, train different models using various machine-learning techniques, and provide the ability to have controlled experiments. Then, some security testing features on cybersecurity aspects, mainly the detection of denial-of-service attacks, were added to the platform. Moreover, the project also tried to establish real-time bidirectional communication between the NDT and the physical network. On the management side, some tasks, such as performance measurements, were also added to the established NDT.

Another approach aiming at the security aspect has been made in [245]. As the number of attacks in the current networks is increasing, it is necessary to train many experts in the field to solve existing issues, such as attack detection and mitigation. However, this training process requires manpower and cost and also includes some margin of error. In addition, there should be some attacks on the network, so the experts can be trained, which is not a reasonable way. In this way, to facilitate the process, there are some cyber ranges based on the NDT that can be employed, and one of the well-known ones called SPIDER [275] has been analyzed in [245]. The authors have tried to analyze how SPIDER can enhance the cyber range by relying on machine learning approaches, here NDT framework, to provide an environment in which experts can be trained. Using NDT in cyber range platforms can provide some advantages, such as automation and better orchestration; as a result, it can improve the security aspect by having more sophisticated professionals and training tools.

Other works on the network slicing aspect are presented in [257] and [288]. The main focus of using NDT for network slicing is to provide clear virtualization of the slices, improve the management of different slices, and enhance their orchestration. In addition, NDT can be used to test different resource allocations on the slices in a reliable experimental environment. In [288] authors proposed an NDT-based network slicing with a focus on the QoS measurement. They proposed an architecture claiming that it can accurately model the slices and the achieved QoS for different applications. Moreover, the proposed NDT could predict some important Key Performance Indicators (KPIs) such as latency. In this way, the combination of the NDT with SDN/NFV techniques could show high dynamism and automation in achieving network slicing goals. On the other hand, [257] tried to use GNN as a tool to create NDT in order to clone the relation between the different slices. Moreover, the virtualized network provided a framework for measuring end-to-end metrics of the slices. The created NDT parameters could be adjusted to mimic the various conditions in the slices; as a result, they can help measure the metrics under different circumstances. The evaluation assessments could show a high accuracy for the created model, leading to metrics close to their real counterparts. In the former work, the main focus was on the 6G network, while in the latter, 5G and Industry 4.0 were the priorities.

Another important enabler for 5G/6G, edge networks and computing [83], and how NDT can be beneficial for its deployment in 6G networks was investigated in [231]. The authors addressed an important question: How can DITEN [231] improve the performance, reduce communication costs, computational load, and the amount of caching? By creating DITEN, the twin features can help monitor various parameters in the network and enhance various decision-making aspects such as routing or resource management. The work could provide a comprehensive overview of DITEN, its different aspects, and potential. The work is a proper guideline on how DITEN can be used in 6G networks and can help both academics and operations.

The design of an NDT and the corresponding architecture to analyze and test Radio Access Network (RAN) for 5G and beyond was done in [251]. The main goal of the architecture was to provide a guideline for building an NDT that can replicate all the elements of RAN and emulate how they function. One of the most appealing parts of the work was the exploitation of RL, which could provide autonomy for the twin to some extent along with an accuracy of 85%. Moreover, the proposed components in the architecture make it possible to collect data from the real RAN, which was one of the reasons for the high accuracy achieved. In addition to the accuracy, the time required to reach acceptable rewards was also low.

As the exploitation of latency-sensitive applications is shaping one of the use cases for 5G and beyond, the feasibility of the NDT deployment for Time Sensitive Networking (TSN) [167] mechanisms over 6G has been explored in [207]. NDT can be an important assistance, as the TSN networks have their own complexity. The authors tried to propose a framework that makes it feasible to create an NDT for both public and private 6G networks. This framework makes it possible for the twin to be trained on collected data from both simulator/emulator tools and real networks. By relying on the framework, a structured NDT can be built that includes different TSN replicated components that are able to interact with each other. Creating an NDT can facilitate addressing the issues that TSN networks have, including 1) Because the cellular and wireless networks are dynamic, TSN configuration is challenging. As a result, the initial configurations and tests can be performed on the NDT, and then, after approval, they can be transferred to the real network. 2) The adverse impacts of the wireless channels, such as distortion, can be investigated within the twin prior to the implementation. 3) The end-to-end management of the TSN networks, which are complex and demanding, can be eased by having the replicated twin.

Optimization decisions are another important factor in cellular communication, which is mostly done by experts in the field. Recently, the combination of machine learning and expert knowledge could improve the process to some extent and bring some autonomy; however, the process still needs human intervention. Another important flaw of the methods is that, due to the human role in the decision-making process, they are not adaptable to the high dynamics of cellular communication. Moreover, the results may be suboptimal or error-prone because humans are capable of making mistakes. One way to address this manual's untidiness is to exploit reinforcement learning, where an agent can learn the network and help to make the process autonomous. Nevertheless, reinforcement learning can encounter issues in dynamic environments, and reaching the optimal point can take a long time or even suboptimal points can be reached. To solve these problems of optimization decisions, a new proposal to combine reinforcement learning, network expert knowledge, and NDT has been proposed in [50] in order to have self-optimization of cellular networks. In this manner, the NDT of the mobile network is created, and then when the optimization decisions are made, the future state of the network is predicted by the twin to assess the new approach. Evaluations through simulations could show that the model achieves even higher accuracy than the experts. This could be done through the comparison of the decisions of the twin and the experts when changing the performance metrics. The results showed that the decisions of the former are more accurate, relying on the reinforcement learning capabilities.

IoV is one of the trending technologies that has gained a lot of attention recently. One of the most important factors in IoV is the amount of real-time traffic, which can play a major role in the quality of the communication. In order to accurately model the communication, an NDT model has been proposed in [86] in a way that provides a framework for connecting vehicles to their virtual representatives via 5G communication. In this way, the administration can use NDT to analyze the traffic data, and based on that, they can change the scheduling and reduce the traffic. The combination of NDT via 5G networks, which they called a digital twin-assisted real-time traffic data prediction, was able to show that changing the scheduling based on the predicted traffic patterns can alleviate traffic congestion. Here, 5G was used as the main backbone for the data collection part of the twin creation, the data that had been provided by the IoV sensors.

The use of NDT for another aspect of cellular communications called Optimization-as-a-Service (OaaS) has been studied in [150], which is a part of the OPTIMAIX project. The framework facilitates the process for third-party applications that intend to perform optimization and planning functions over 5G/6G networks in order to have sophisticated network service and resource allocation. This can be done by relying on the data-driven NDT creation, which makes it possible for the third parties to run the approaches in the twin before the implementation. In this manner, first, the OaaS architecture was analyzed in detail, then different implementation scenarios were investigated, and finally how NDT can be built and used to improve its mechanism was presented. It was shown that by combining NDT and OaaS, the quality of the network services and resource allocation can be improved.

QoE is another important factor in the 6G network, as it promises to deliver high values. The authors in [297] analyzed the QoE, but the other way around. Instead of the investigation of the QoE for 6G, which has been done in many studies such as [148], they analyzed the DT service provided in 6G networks. To do so, they have created a dataset to create the NDT that can be used to analyze the delivered QoE, which is called Human Digital Twinss (HDTs). The main focus of the created NDT is to analyze and monitor the parameters that are related to the QoE for users in the 6G networks and how well this twin works. The creation process included a framework for the HDT that incorporates all the necessary components, such as the twin itself, encoders, base stations, the 6G transmission networks, QoE monitors, decoders, and users. Once created, they validated the twin by using simulation environments under 60 GHz frequency, which showed the high accuracy of the model. This framework can be a great tool as an enabler for improving the user experience in 6G networks.

The same procedure, but targeting QoS in 6G, was done in [236]. The main leverage for the creation of the twin was on the SDN; as a result, the novel proposed framework was called digital twin function virtualization (DTFV). The proposed architecture for the framework consists of a physical network and a twin, where they have bidirectional communication to collect data and provide feedback to the network. The novelty is to have a layer on top of them both, using SDN techniques, where the SDN controller is the main brain, to improve the service response quality and enhance the adjustment of the virtualized digital twin resources scheduled. For the evaluation, a 6G network including vehicles, edge networks, and the corresponding twins was established. Then, the comparison of the new model with three heuristic methods showed a higher accuracy for the DTFV.

There has been some work visioning the future direction of NDT for future cellular networks, discussing and analyzing the requirements, architecture, trends, and visions in [62], [104], and [238]. They all tried to show the roadmap for building NDT for future cellular communications, particularly 6G, in a way that can improve resource planning, management, and control. In [62], the vision for the European 6G-TWIN consortium, whose main focus is to create a virtual platform that communicates with its physical world counterpart through the integration of NDT AI-based 6G architecture, and the existing challenges were presented. In the network phase, the goal is to improve the user experience by virtualizing the network and its complex aspects to facilitate the work on the operational side. Moreover, various parts of the 6G-TWIN, including the physical network, time management service, energy-saving demonstrator, network applications, and orchestration of AI-based network functions, functional models for network planning, management and control, and more, were also explored. Providing the high-level architecture of 6G-TWIN along with the analysis of various, challenges, and AI-native components could provide a great guideline for those who are interested in working NDT towards 6G networks.

Similar works to [62] have also been done in [104], [238]. They have also tried to investigate various challenges in creating an NDT for 6G networks. The former tried to envision the key requirements of 6G NDT, provide an architecture for it, and analyze various trends such as edge-based twins, cloud-based twins, and edge-cloud-based twins. In the first step, they provided the basic architecture, then analyzed the benefits, and finally looked at various features such as scalability, elasticity, and mobility. One of the most important takeaways from the work is the emphasis on the role of the NDT as a key enabler for 6G services. The authors in [238] have also tried to propose an architecture for 6G NDT with a focus on smart resource management and intelligent service provisioning. The goal was also to answer the question of how we can create a data-driven virtualized clone of the 6G network to aid in the investigation in order to improve the traffic management aspects of the network. The main difference in the work was the use of reinforcement learning-based approaches to move towards having the 6G NDT. Table 4 includes a summary of the covered works.

The iterative learning process of FL poses significant challenges to mobile networks. FL relies on a fast communication system to ensure efficient and successful training. However, this requires a stable set of participants with sufficient connectivity for precise and consistent model training, which is difficult to achieve in dynamic environments. This challenge becomes particularly critical in networks with high user mobility, such as vehicular networks. To address this issue, a lightweight model training framework is proposed for ad-hoc mobile network environments [297]. This framework, known as the end-edge-cloud structured three-layer Federated Reinforcement Learning, integrates a cloud-edge DT arrangement. The proposed scheme enables a collaborative supervisory mechanism that effectively optimizes client node selection and global aggregation frequency during the FL process, thereby enhancing performance in dynamic and mobile scenarios.

Moreover, Federated Learning performance is often hindered by core network traffic congestion and the limited computational capacity of user devices. This distributed learning paradigm faces significant latency and congestion challenges, especially when communication occurs over long distances between user devices and cloud servers. To address these challenges, the proposed solution by [84] leverages a DT network to create a digital model of the system. It incorporates edge servers to mitigate the straggler effect on user devices in FL processes. The framework is tailored for Heterogeneous Cellular Networkss (HCNss) and integrates Multi-Access Edge Computing (MEC) with DT to alleviate core network congestion. The framework also introduces a dynamic mechanism to manage user-device associations and resource provisioning, further enhancing the efficiency and reliability of FL-based operations.

The concerns of security and privacy in 6G networks arise from the use of ML and AI schemes, particularly in relation to data protection. To address these issues, a solution that optimizes critical 6G network metrics by leveraging ML and AI techniques, incorporating transfer learning and blockchain to maintain privacy and security during network expansion and resurgence was done in [172].

To sum up, 5G/6G networks are expanding at a rapid pace and will dominate the everyday lives of human beings. This is because they offer many novel use cases and features such as high data rate, low latency, AI deployment, edge computing, and more, which can enhance the quality of communication. However, due to many issues such as complexity, high dynamics, rapid expansion, and a lack of simulation/emulation tools, it is hard to test a novel approach for the networks before the implementation. To address this issue, NDT can be used to clone the mobile network and provide a framework for detailed analysis. As a result, this subsection has covered various works in NDT towards 5G/6G in detail.

B. NDT AND INDUSTRIAL IoT

Research in [234] indicates that about 18% of DT's manufacturing applications are focused on the design space, followed by production areas (35%), health management and prognostics (38%), and other areas (9%). Well-known DT applications for smart manufacturing include product design, production scheduling, problem detection, and predictive maintenance [93], [210], [233], [258], [277], [295], that are intended for small-scale real physical systems. In most manufacturing DT applications, the latency of data transmission between the physical system and the DT can be decreased by placing the DTs close to the physical systems. In addition, small-scale physical systems usually have low data volumes, which results in manageable data processing delays at the DTs. therefore, the DTs and their physical systems may be precisely synchronized, which is very crucial for optimizing production processes. The DTs utilized in mobile computing offloading are likewise intended for small-scale physical systems, including edge servers, base stations, and end devices [141], [142], [255]. For these DTs, short delays in decision-making are necessary to facilitate resource allocations, computation offloading decisions, and safe and reliable computation in edge networks.

The IIoT is a complex system due to its restrictions as it controls hard real-time applications [32]. NDTs are essential to the IIoT for supply chain logistics and manufacturing process optimization. They give producers an in-depth, real-time view of their production lines, enabling quick adjustments to boost productivity and quality [278]. Preventive measures reduce downtime and increase equipment lifespan by predicting machine faults before they happen; this is a capability made possible by NDTs. DTs can be used to model logistics networks in supply chains to find bottlenecks and improve routes and inventory levels. This feature is extremely beneficial in intricate international supply chains, where real-time modifications can result in substantial cost savings and improved responsiveness to market fluctuations. Authors

TABLE 4. NDT as an enabler for 5G/6G networks.

Ref.	Summary
NDT for 5G and 6G [168], [175]	Analysis of the NDT from an architectural and beneficial perspective for 5G and 6G.
Modeling and deployment methodology of NDT for 5G networks [199]	A guideline on how to use NDT to model a 5G network's various characteristics, such as communications, devices, links, applications, and operating environment.
NDT for 6G networks [78]	Analysis of requirements, key technologies, and the corresponding issues in creating an NDT for 6G networks.
B5GEMINI project [101]	A NDT for 5G/6G aiming at security and management aspects.
Analysis of the SPIDER cyber range [245]	How cyber range machine-learning tools such as SPIDER can help train security experts.
NDT as an enabler for network slicing [288], [257]	Improving dynamism and automation in network slicing by relying on NDT.
DITEN toward 6G networks [231]	A comprehensive analysis of DITEN various aspects.
Building NDT for RAN [251]	Proposing an architecture for creating NDT for RAN for 5G and beyond.
NDT for TSN network in 6G [207]	Proposing a framework that makes it feasible to create an NDT for both public and private 6G.
Optimization decisions improvement [50]	The combination of reinforcement learning, network expert knowledge, and NDT to provide an autonomous self-optimization decision process for mobile networks.
Using NDT via 5G for the IoV [86]	Creating an IoV-based NDT by using collected data from vehicles' sensors through 5G.
NDT for OaaS over 5G/6G networks [150]	Simplifying the process for the third-party applications that intend to perform optimization and planning functions.
An NDT to assess user experience in 6G networks [148]	Creating an NDT based on SDN techniques to analyze and improve QoS in 6G networks and support innovative 6G services
Twinning the QoS in 6G [236]	the creation of an NDT that can be used to analyze the delivered QoE in 6G networks
NDT for 6G networks, a vision [62], [104], [238]	Providing a roadmap and future directions for building NDT for future cellular communications, particularly 6G
Federated Learning faces challenges in mobile networks due to dynamic environments [297]	A three-layer Federated Reinforcement Learning framework is proposed to optimize client selection and aggregation frequency, improving performance in such scenarios.
Federated Learning in Heterogeneous Cellular Networks [84]	Integrates Digital Twin networks, Multi-Access Edge Computing, and dynamic resource management to reduce congestion and improve efficiency.
addressing security and privacy concerns in 6G [172]	using ML, AI, transfer learning, and blockchain to optimize network metrics while ensuring data protection.

in [46] reviewed the applications of DT in supply chain management and positions, particularly focused on DT's role in asset visualization, operations traceability, transport maintenance, remote assistance, and design customization. To facilitate intelligent real-time administration of IIoT networks, the authors in [106] created an NDT for the IIoT in which communication infrastructure, sensors, and actuators are all copied in the DT. In their approach, new networking services can be effectively integrated and utilized throughout the network life-cycle. Such services include network diagnosis, energy optimization, predictive maintenance, and resource allocation. The authors provided a prototype implementation to validate their proposed design.

In [107], a holistic NDT architecture was proposed for IIoT to allow closed-loop network management across the whole network life cycle. This architecture adopts the SDN concept as an expression of network softwarization. The SDN controller permits establishing the association between each DT of the industrial system and its physical equivalent. The authors verified their architecture's viability by selecting the best communication mechanism to meet a Flexible Production System's real-time requirements. From another perspective, it is crucial to achieve the economical, intelligent, and energy-efficient deployment of 6G networks in smart industries. For managing complex and varied data, a mobile-enhanced edge computing-cloud collaborative system and a DT-based system architecture were suggested in [276]. Additionally, a method based on DT and AI was created to allow the intelligent planning and implementation of 6G networks in industrial units. This method results in lower operating costs and improved network performance.

From another viewpoint, an NDT can forecast network bottlenecks and congestion by examining real-time network traffic and topological data. To attain business equilibrium, Tang et al. [232] presented an NDT-assisted IIoT Network Slicing (NDT-IIoT NS) architecture. To provide highly customized IIoT services, they created a DT-assisted resource allocation model and designed an optimization problem to optimize the equilibrium rate's weighted net profit. For obtaining service equalization in NDT-IIoT NS resource allocation, the authors presented a dual-channel weighted (DCW) analysis network. To boost convergence speed, they also included a corresponding improved Prioritized Experience Replay (PER). They also suggested a multi-agent deep deterministic policy gradient technique for resource allocation in NDTIIoT NS (DCW-PER multi-agent) supported by distributed DTs. The simulation results show how well their algorithm works to satisfy service requirements, achieve service balance, accelerate convergence, and get around DCW networks' sluggish convergence speed.

Task offloading poses significant challenges to the performance of next-generation networks, as it often overlooks critical factors such as network heterogeneity, IoT device mobility, and the diverse requests of IoT devices. To mitigate these issues, an optimized approach is proposed that addresses key aspects, including IoT device association, service DT

TABLE 5. NDT in IIoT.

Ref.	Summary
[46]	It reviews the applications of DT in supply chain management and positions.
[106]	It proposes the creation of an NDT for the IIoT in which communication infrastructure, sensors, and actuators are all copied in the DT.
[107]	It proposes a holistic NDT architecture for the IIoT to allow closed-loop network management across the whole network life cycle.
[276]	It proposes a mobile-enhanced edge computing-cloud collaborative system and a DT-based system architecture.
[232]	It presents an NDT-IIoT NS architecture and a DT-assisted resource allocation model to optimize the equilibrium rate's weighted network profit.
[135]	Task offloading challenges next-gen networks by overlooking network heterogeneity and IoT mobility.

placement, and the allocation of network and computing resources [135]. By enabling a partial offloading mechanism, this solution significantly reduces a critical metric: the DT job completion time for all IoT devices, thereby improving overall network efficiency and performance.

Even though NDT has many advantages for industrial IoT applications, there are still certain difficulties when using this technology in the manufacturing domain. Such challenges are [60], [274]:

- Co-channel interference from other equipment frequently compromises information exchange across physical and virtual environments in industrial settings, resulting in errors and missing data. One of the main challenges is figuring out how to build realistic DT models of the manufacturing plants with inaccurate data.
- Several industrial facilities collaborate under the direction of NDT in contemporary automated production lines. These facilities have intricate and erratic linkages when scheduling production and ensuring the quality of the final output. In this intricate situation, it is challenging for the NDT to create a matching twin model and develop an ideal management plan.
- Protecting personal information during the creation of DTs in industrial manufacturing is another difficulty. Core data (e.g., production parameters and user personal information) are needed to build and update the NDT. Meanwhile, private information may be disclosed to malevolent attackers throughout the information delivery process, particularly when using wireless connections. Therefore, another problem that must be resolved is guaranteeing privacy in the NDT operation.

Table 5 highlights and summarizes some important research works on NDTs in IIoT.

C. NDT OVER SMART CITIES

A smart city uses data and digital technologies to augment residents' quality of life, improve experiences, support sustainability, and boost economic development within urban areas. It envisions and supports conditions where necessary domains of the urban infrastructure such as smart transportation, smart healthcare services, environmental monitoring, intelligent infrastructures, smart grids, safety, and emergency response are seamlessly integrated. The ease with which smart city features and functions can be blended with DT technology is made possible by this integration [92]. Smart cities aim at quick disaster/mishap mitigation, quick assistance in emergencies, lowering operational costs, better decision-making competence for enabling superior governance, and supporting commercial uptake and development for residents' welfare. These objectives have resulted in the rapid development of DTs in the context of smart cities [164].

Zurich and Vienna are two indicative smart cities that have deployed their digital twins. For Zurich, a DT [213] was created by converting 3D spatial data and city models into a virtual environment. The city models include structures, bridges, vegetation, and other elements. In particular, the authors presented how ML techniques can forecast the effects of urban climate by using data on air quality and current weather. Similarly, to build a living DT using AI techniques, the Vienna city digital geoTwin [118] is connected to municipal data, including social-economical, power usage, and maintenance administration data.

Usual procedures and simulations in smart cities are being replaced by well-designed DT-based dynamic systems. For example, Fan et al. [61] proposed combining artificial and human reasoning for a disaster city digital twin. The authors [164] argued that in the ever-changing smart city environment, with the help of advanced DT technologies and cutting-edge DT-based smart city deployments, several use cases and applications are beginning to look like a conceivable DT-based system. Such application scenarios are traffic flow analysis (i.e. understanding the impacts of variations in traffic flows within metropolitan regions on inhabitants' mobility), mobile alerts (i.e., sending warning notifications dynamically to citizens about water-level or fire hazards, evacuation orders, and safety guidelines), and so on. These scenarios highlight the potential of combining smart city apps with DTs to supply vital facts and critical information to authorities to make outstanding domain-specific decisions for the well-being of citizens.

Nowadays, DTs are increasingly being used in smart cities to plan the disposition of intelligent infrastructure, anticipate real-time interactions between citizens and systems, understand how these interactions are interdependent, help communities in need, and modify administration regulations [244]. A DT offers support for digital models for real-world tangible objects and systems. Additionally, DT-based systems focus on the ability to sense, transmit, receive, and process data flows through distributed IoT systems effectively. In practice, the method revolves around infiltrating the city's DT with real-time data gathered and recorded through urban information systems and IoT infrastructure. A well-designed DT makes it possible to anticipate changes in urban infrastructure by adeptly evaluating the data acquired and gathered in real-time regarding the interactions, correlations, and variations between citizens and systems. In addition, data analysts can employ the DT to uncover answers to "what-if" queries, which helps them understand and forecast how smart cities will function in specific environmental, social, and economic contexts. Moreover, this also aids in identifying circumstances and factors that may lead to failures [92]. Furthermore, the more real-time data we collect through IoT-based infrastructures integrated into sensitive services inside a city, the more research possibilities will open up for creating advanced AI/ML schemes [67]. In smart cities, IIoT and IoT are two excellent paradigms that easily fit and suggest suitable assistance between real-world and virtual twins, to integrate reality and virtuality. The eventual benefit of these paradigms is their competency to collect real-time data from a wide range of heterogeneous sources over various dissimilar communication channels to aid data analytics through distributed computing and collaborative frameworks [87]. However, the most fundamental feature of these paradigms is the use of sophisticated IIoT/IoT devices, such as intelligent wearables, smart sophisticated environmental sensors, GPS trackers, RFID tags, and proximity sensors, which are exceedingly convenient to use and are relatively considered as economic data sources that can reflect the expressive shadow of the reality that the deployed cloud-based DT can understand and examine. Furthermore, in the case of smart cities, both paradigms can help accelerate the creation of DTs by offering an appropriate platform that can explore and analyze a wide range of data formats from diverse protocols. The specialized IoT/IIoT-based devices are typically designed keeping some specific communication standards into consideration, such as 5G-based networking features (i.e., uRLLC, eMBB, and mMTC), MTConnect [87], CoAP [26], and MQTT. Both communication paradigms bridge the gap by connecting the above-stated standards with the DT [157]. The authors [157] emphasized the significance of selecting sophisticated IoT/IIoT devices, accompanying IoT platforms, and communication protocols in affecting DT characteristics. One of the most important characteristics in this context is the *synchronization rate*, which is eventually determined by the DT's use case. For instance, in latency-sensitive smart city applications (e.g., traffic management, public safety and surveillance, autonomous vehicles, emergency response systems, remote healthcare, and so on), it is vital to choose the suitable communication protocols while the communication link (between two communicating parties) may employ secure uRLLC. Meanwhile, in the context of applications where latency is not an important issue (e.g., in historical data aggregation, urban planning, and analysis tools, non-critical environmental monitoring, and so on), this requirement is not so necessarily rigorous. Overall, there is a shortage of unified models and data fusion frameworks in the literature, which requires extensive effort in this area. In addition, this issue can substantially challenge the constructive apprehension of DT technology by cities that struggle to manage and tackle several urban obstacles with sophisticated data-driven solutions. The right standards for DT implementation have been defined thus far, but not much has been done in terms of a unified model and fitting interactions of data fusions concerning digital and physical data exchange. This is true in smart cities, where the DT-based paradigm is just now beginning to gain momentum. Subsequently, to narrow this gap, the contributors in [188] presented a novel framework and unified model for digital city twins. Their model and the framework enable appropriate data exchange of several DT models through suitably designed APIs. Specifically, their framework serves as a holding area for information, simulations, and models that eventually interact directly in a given setting, while it offers beneficial information for authorities to make efficient decisions in smart cities. Recently, the notion of DTs has been extensively applied to intelligent residences [36], [37], smart cities, [51], [57], [86], [160], [188], and healthcare systems [58], [196]. While this DT nomenclature has been integrated into many services, its adoption in the networking context is still comparatively new and has not been thoroughly investigated. However, since the emergence of this terminology and expertise, it has evolved in tandem with multiple advances in a variety of technologies that include networking systems, as previously reported by the authors [190], [274]. Nonetheless, it has been presented that creating and implementing a fully viable DT for complex systems is still a long way off and necessitates careful and detailed examination. Since the core premise of DT is that full renderings of a physical item or system should be indistinguishable from their genuine equivalents, attaining this in the real world is arguably exceedingly difficult, though not unachievable. In most cases, a completely operational DT realization must imitate all the core features of its physical counterpart, as well as be capable of reflecting all of the attributes that correlate to its physical system that an application requires. To address this issue, suitable networking capability is required for subsequent DT implementations [244].

Table 6 summarizes some important works on NDT in smart cities.

D. NDT IN INTELLIGENT TRANSPORTATION SYSTEMS

Intelligent Transportation Systems (ITS) have significantly improved the safety, planning, cost-effectiveness, and sustainability of modern transportation networks. The current advancements [14], [73], [97], [290] in ITS technologies, like real-time data analytics, ML algorithms, and IoT devices have a huge impact on traffic management, public transportation, and vehicular communication systems. Such advances aim to reduce traffic congestion, minimize environmental impact, and improve road safety. The industrial implementation of

TABLE 6. NDT over smart cities.

Ref.	Summary
[213]	Zurich's DT includes ML-based techniques which forecast urban climatic conditions and subsequently assist in understanding their effects on urban planning.
[118]	geoTwin is associated with important Vienna's data, such as power usage, socioeconomic, and maintenance administration data.
[61]	It presents numerous possible areas of research relevant to the use of AI particularly in the case of disaster management.
[188]	DUET: It is a novel framework and unified model for the case of digital city twins. It enables data fusion and integrates typically for the case of physical and virtual information communication in a smart city DT environment.
[170]	It presents a city-scale DT prototype for supporting policy designing to address some imperative urban environment issues such as growth administration, congestion, air pollution, and the restricted capacity of the local power infrastructure.
[160]	It highlights the changing states in spatiotemporal flux and underlines the need to comprehend them to sustain growth.
[203]	mySmartLife: A novel framework which utilizes the concept of IoT-based infrastructure in cities to model a smart city DT.

autonomous driving necessitates a substantial amount of work in the areas of planning and control [90]. Consequently, an experimental procedure is typically carried out in a high-fidelity simulator, before being implemented in a real-world setting. This can save the expenses associated with research and development. Various techniques have been suggested to bridge the gap between simulations and reality [100], [242].

NDT in traffic engineering [19] offers a dynamic, real-time virtual representation of real traffic networks. These digital models enable traffic engineers to simulate, predict, and enhance traffic flow and infrastructure. Applications include enhancing traffic management systems, improving congestion prediction, and facilitating the design and testing of new traffic control strategies. Also, using DTs, traffic engineers can assess the impact of various scenarios, such as accidents or construction, without disrupting actual traffic, leading to more efficient and safer road networks [57]. An NDT can achieve real-time vehicle tracking by continuously monitoring its driving conditions and assessing traffic scenarios [187]. This skill is useful to achieve proper active traffic flow management in an urban context. In [94], [126], the authors emphasized utilizing an NDT for vehicle safety by predicting potential risks. These studies aid in developing strategies to avoid accidents and other complications before they occur by anticipating potential threats, especially in changing and busy traffic situations. Many authors [12], [138], [254] have emphasized the use of the DT idea regarding autonomous driving, resulting in anticipating the likelihood of a vehicle crash through the use of data collection and analysis, thereby confirming an innocuous driving scenario for all road users. The motivation behind implementing an NDT in intelligent autonomous systems lies in its ability to simulate and test network conditions, optimize performance, and enhance real-time surveillance and predictive maintenance. By creating a digital "image" of the network, an NDT enables cost-effective risk mitigation, resource allocation, and performance tuning, ensuring efficient and reliable operations. An NDT facilitates scalability, flexibility, and security through continuous data-driven insights, ML integration, and vulnerability assessment. Ultimately, NDTs drive innovation, regulatory compliance, and system resilience, making them indispensable for advancing intelligent autonomous systems.

Even though DTs aim to replicate Cyber-physical Systems (CPSs), there are many challenges in DT modeling. Combining and linking huge amounts of data from heterogeneous physical sources makes it difficult to create ideal network replicas in real-world traffic management [22], [53], [67]. To solve real-time traffic problems, a DT-based ITS can use various technologies, such as Fused Millimeter-Wave Radar, Video surveillance cameras, Holographic Perception, etc. [39], [174], [186], [265]. Combining these technologies with DT can efficiently reduce the waste of traffic resources, uncertainty of traffic events, and traffic signal system functionalities.

In such context, the authors in [137] focused on Virtual Reality (VR)-based IoVs from the perspective of DT with two major issues: data distribution technologies and user applications. This work is beneficial for the advancement of Transportation Digital Twin (TDT) which can deal with real-time traffic systems.

In light of the above research, the authors in [89] portrayed a comparative study of DT, Simulation to Reality (Sim2real), and parallel intelligence. Unlike Sim2real, a DT in the context of an ITS refers to a virtual model of a real transportation network that is continually informed with real-time data from the actual system. This digital representation enables simulations, analysis, and optimization of transportation networks, providing insights into traffic flow, infrastructure status, and potential improvements. DTs are commonly employed in multi-scale environmental and vehicle simulations. Using real sensor data, data interaction between the driving data analysis model and virtual reality is used to plan the motion of the twin body and real car in the virtual scenes. Through interactions between the virtual and physical worlds, the DT system continuously improves the model's accuracy.

In [30], a DT prototype was proposed based on CPSs for automated vehicles. This research highlighted the strong connectivity of CPS, Big Data, and DT technologies in the era of Industry 4.0, which is the most beneficial aspect of autonomous decision-making and self-adaptation in a system.

In [191], a DT system for self-governing electric vehicles was suggested. In this work, the authors showed how the ISEAUTO [214] car's sensors and vehicle models are modeled using MATLAB, and how ML software is used to merge data from the real devices and virtual sensors.

In [259], the authors introduced a new platform, called the Smart Mobility Digital Twin (SMDT), designed to manage Connected and Autonomous Vehicles (CAV) over next-generation wireless networks. The SMDT platform enhances the autonomous driving experience by leveraging DT capabilities by integrating cloud services. DT data was used to create a revolutionary navigation system that increased traffic efficiency and safety. The navigation system was implemented using cutting-edge technologies such as CAVs, Road-side Units Road Side Unit (RSU), cloud computing, and Cellular V2X (C-V2X).

For a DT-based Intelligent Vehicular System (IVS), a novel resource scheduling scheme (based on parallel intelligence) was proposed in [286]. The scientists modeled an adaptive hybrid particle swarm in conjunction with a genetic algorithm to simultaneously optimize load balancing, resource allocation, and the offloading mechanism while many vehicle-dependent tasks are offloaded to multiple computing platforms.

In [266], a novel Mobility Digital Twin (MDT) framework was proposed as an AI-driven setup for edge-device, adapted for mobility services. MDT comprises three primary components in physical space (human, vehicle, and traffic) and their corresponding DTs in the virtual realm. To facilitate the digital activities of the proposed framework, including modeling, learning, simulation, storage, and prediction, a cloud-edge architecture is built using Amazon Web Services (AWS). A case study on Personalized Adaptive Cruise Control (P-ACC) exemplifies the integration of micro-services from all three digital components: (i) human DT for user management and driver classification, (ii) vehicle DT for cloud-based advanced systems that assist drivers, and (iii) traffic DT for traffic flow monitoring and variable speed limit control.

In [112], an innovative strategy was introduced for Virtual Vehicle (VV) models which includes both driver and vehicle in the virtual state. Moreover, an Recurrent neural network (RNN), using recent technologies such as a reliable virtual vehicle model, ML, IoT edge analytics, and 5G communication, was designed by the authors. These advanced technologies are integrated to mitigate traffic congestion issues effectively. The paper provides a clear and structured explanation of how these advancements in DT technology work together to address common causes of traffic congestion. This work also addressed the insights into the practical applications related to improved urban mobility and reduced traffic disruptions.

In [298], [299], an Intelligent DT-based Software-Defined Vehicular Network (IDT-SDVN) was proposed which alleviates the limitations of traditional SDVN, like centralized control, high training time, dynamic topological changes, etc. The main achievement of this scheme is that it improves end-to-end communication delay, shortens the path length, and gives a better packet delivery ratio (PDR). In a DT-based hierarchical routing approach, proposed in [298], the authors described their approach in four phases: (i) *policy training*, where multiple agents in virtual DT networks learn and compute various policies parallelly; (ii) *policy generation*, where learned policies are combined to create new policies tailored to complex communication needs; (iii) *policy deployment*, where the best policy is chosen based on real-time network conditions and message types, and (iv) *relay selection* to decide relay vehicles through a hop-by-hop process throughout the chosen path.

A model-free DRL-based MDP model was formulated in [8], which can overcome the complexity of dynamically varying topology in multihop scenarios in IoV. DT aims at monitoring topology heterogeneity in dynamic vehicular scenarios and using fewer resources during packet transmission, which increases robustness over vehicular mobility. In this work, the authors use the notion of Multi-agent-based Deep Deterministic Policy Gradient Method (MADDPG) with the Lyapunov drift-plus-penalty function (LyMADDPG) for each agent's training for policy evaluation. Their approach integrates edge AI and DT, which achieves maximized channel gain and minimized packet delay in comparison with traditional MADDPG. The limitations of DT in Vehicular Ad-hoc Networks (VANET)-assisted ITS, due to its huge data and varying features in real-time, are highlighted in [169]. To mitigate these challenges of the time-varying traffic flow with huge fluctuations, the researchers developed a novel network travel prediction methodology based on DQN.

Autonomous Vehicles (AV) offers several benefits in terms of safety and security, including a decrease in collisions and the preservation of a cautious driving and pedestrian environment. In this context, the shift to data-driven cars is linked to the idea of a DT, particularly when it comes to AV design. Recently, various applications have been created to demonstrate the effectiveness of TDT systems in improving the safety and mobility of current transportation systems [91], such as cooperative driving [265], cooperative ramp merging [128], and modeling of driver behavior [44], [112]. As a result, the industry has a strong incentive to research how cyber-attacks affect the car sector and create new platform solutions for anti-virus software that prioritizes data privacy, safety, and security [166].

A DT-based safety and security-based approach for autonomous vehicles was proposed in [12]. Here, the authors explored DT techniques for the automatic decision-making system using sensor data collected from AVs and consecutively sending the decision to them. An application based on DT was created by Chen and colleagues [44] to prevent car crashes. To anticipate drivers' actions when driving, their method first employed DTs of drivers. Following that, a risk analysis model was put into place that calculated the predicted driver behavior generated by the shared DT models. A prototype of Advanced Driver Assistance System (ADAS), based on DT, was presented in [7], which highlights vehicle safety concerns. For better safety measures, many researchers highlighted human driver modeling using DT technology. In light of this evidence, authors in [112] developed a DT-based model to predict driver intention. They conceptualized the vehicle at the virtual level, which consists of the vehicle and driver. With a similar notion, [44] they modeled a framework with a DT of drivers, which is shared among drivers of connected vehicles, aiming to increase safety among them. In [146], a Cooperative ITS (CITS) system (based on the DT model) was proposed as a solution to security problems in the DL environment. The proposed algorithm minimizes the delay of data transmission, increases accuracy, and also minimizes the expansion of traffic congestion in smart cities. Also, a comprehensive study regarding the Transportation DT in terms of the requirements, underlying architecture, hardship, and future scope has been elaborated.

In an autonomous driving context, numerous advanced machine-vision-based algorithms and procedures are utilized to extract detailed traffic flow complex information [131], [254]. Meanwhile, numerous advanced ML-based schemes are employed for assessing critical factors (such as trajectory, speed, etc.) and semantic segmentation [131], [303]. Further, various physics-oriented dynamic models (i.e., geometric models) are considered swiftly by researchers for commendably implementing digital-virtual layers [254].

The authors [254] provided a DT-based scenario that includes realizing vehicle motion in the setting of an expressway. The authors performed a data collection technique that focuses on gathering real-time crucial data about vehicle movement patterns on an expressway. The approach then advises by creating a virtual model that simulates real-world vehicle motion scenarios. As a result, it allows observation of patterns in vehicle movement activities and, eventually, the prompt capture of cars that pose increased driving risks. The authors in [94] investigated the existing models in the literature and improved the model by analyzing the associated critical factors typically for the case of urban road networks. The authors designed a DT-based road network to analyze the traffic pattern and traffic's associated operations effectively. This aids the model in understanding general traffic patterns and congestion without concentrating on individual vehicles. Furthermore, the authors used both the Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) paradigms, because CNNs are useful for processing spatial information such as road-infrastructure images and layouts. In contrast, LSTMs are good at handling temporal sequences such as changes over time. Notably, data bias (during information aggregation for a DT) can certainly lead to imprecision in the digital-virtual model. This emphasizes the importance of rigorous data processing to guarantee that the model accurately and dependably represents the real-world system. Proper data management, including abolishing any biases and validating the model, is precarious for the sustained efficacy of DT technology. Table 7 summarizes some important works on NDT in ITS.

To gain a clearer understanding of the latest advancements in applying the NDT paradigm to next-generation communication networks, Table 8 summarizes the most recent contributions in the field. This table provides valuable insights into the progress of NDT applications in telecommunications and offers guidance for shaping future research directions. It highlights the evolving trends, key research areas, and innovative approaches in integrating NDT with telecommunications systems. By presenting these recent developments, the table not only offers valuable insights into the progress of NDT applications but also serves as a guide for identifying emerging challenges and opportunities, helping to shape the direction of future research and development in this area. Ultimately, these insights can help optimize the implementation of NDT in telecommunications, improving network efficiency, performance, and resilience. Through these efforts, the potential of NDT to transform next-generation communication networks can be fully realized, leading to smarter, more adaptive, and future-proof networks.

In the next section, we investigate how ML and DL methods can be integrated with other technologies to face various situations that may occur in an NDT such as anomaly handling, system state monitoring, resource allocation, task offloading, and model optimization.

X. COLLABORATION OF DT AND ML METHODS WITH NDT

The high complexity of current networks presents challenges in the design and implementation of NDTs. Real-time modeling of physical entities, managing large amounts of data via IoT and IoV, optimizing system performance with limited resources, and ensuring model security are all essential needs of any system. The constantly changing environment of networks creates potential challenges for effectively managing data processing, data communication, and user operations.

Recently, ML-DL collaboration has been combined with the concept of NDT. Such a collaboration holds great potential across various complex applications in IoT/IIoT environments and beyond. Moreover, the ML-DL collaboration suggests a prevailing boost in NDTs' complex operations, which aids in managing a wide range of issues associated with complex network applications. Different ML and deep learning techniques offer powerful data processing and application capabilities, enabling an NDT to detect detailed patterns and relationships in large datasets [184] NDTs can also increase system performance by processing multimodal traffic flow data and automatically capture nonlinear linkages and spatiotemporal interdependence. Due to the enormous significance of ML and DL in the context of NDT, many researchers considered the collaboration of novel learning techniques, such as FL [294] and transfer learning [260] with the DT model. In an ML-enabled NDT system, multiple entities can collaboratively interact with others to achieve a common goal. These entities comprise physical components,

TABLE 7. NDT in intelligent transportation systems.

Ref.	Summary
[290]	An ITS deals with and oversees manual and self-driven cars. It solves any potential difficulties while interacting with autonomous vehicles, most notably those involving accurate route prediction and object identification. Additionally, self-driven vehicles employ exchanging critical time-sensitive data via signals and adapt according to the situation, whereas humans driving manually act more suitably as per the situation. This work presents a DL-model-based traffic safety scheme, especially for the scenario of 5G-enabled ITS, and highlights these issues when dealing with such a dynamic environment where heterogeneous vehicles are involved.
[57]	This work investigates the employability of DT conception of the road infrastructure by deploying a DT box to the roads that contains a number of sophisticated IoT devices and a 3600 camera linked to a single onboard computer. By repeatedly sending vital real-time road infrastructure data to the edge or cloud, such as GPS location, continuous live streaming, estimated humidity, and temperature measurements, this used DT box creates a DT of a road infrastructure. Additionally, an object recognition method is included in the suggested model to separate all likely objects from the recorded video stream.
[94], [126]	These works have highlighted the use of an NDT for vehicle safety by anticipating potential hazards. By foreseeing potential dangers, particularly in dynamic and congested traffic settings, these studies assist in the development of solutions to prevent accidents and other issues before they occur.
[12], [138], [254]	These works have highlighted the application of the DT principle to autonomous driving. This has led to the anticipation of the possibility of a vehicle crash through data gathering and analysis, validating a driving scenario that is safe for all users of the road.
[137]	This work enables dynamically generated data processing particularly for the case of DT-based IoV systems. It concentrated on IoVs based on VR from the standpoint of DT, addressing two key concerns: user applications and data delivery systems. The development of TDTs, which can handle real-time traffic systems, will benefit from this effort.
[30], [191]	These works presented the DT prototype as well as DT systems for the case of autonomous vehicles. These studies emphasized the most advantageous feature of autonomous decision-making and self-adaptation in a system: the robust interconnectedness of CPSs, Big Data, ML, and DT technologies.

their virtual counterparts, different services, ML-based data engines, and an interconnecting network for communication among them. ML-based technologies act as an enabler for NDT by providing efficient support of data collection, cloud and edge intelligence, optimized communications, safety, and security in computer networking and other industries, like smart transportation, manufacturing, healthcare, defense, and so on. Table 9 summarizes the integration of NDT with ML.

A. ML-DL COLLABORATION WITH NDT FOR ANOMALY HANDLING

One of the aspects of an ML/DL-based framework of NDT is to address the requirements for anomaly handling, such as anomaly detection, fault detection and maintenance, intrusion detection, and anomaly monitoring in complex networks [187]. The learning methods improve NDT's efficiency in real-time data collection and continuous monitoring, leading to more accurate and responsive anomaly detection operations. In [122], [125], the authors proposed an advanced NDT system to identify anomalies in fixed broadband services. Their model can detect anomalies in multi-layer network topologies and process multi-dimensional data to detect quickly any quality deterioration in network access. The main advantage of ML-driven techniques in NDT is to detect the fault in initial states so that system performances are not hampered [272]. The major application of such techniques includes IIoT. In [88], the authors proposed an IIoT-masked one-dimensional convolutional autocoder (MOCAE). Another application of ML collaboration with NDT is networking. In [262], the authors used advanced ML and DL techniques to monitor and in real-time detect unusual faults within networks that use Network Function Virtualization (NFV). These techniques help to understand how different faults are interrelated and can improve the management and reliability of such networks.

B. ML-DL COLLABORATION WITH NDT FOR SYSTEM STATE MONITORING (SSM)

In this case, the objective of NDT is to keep a continuous monitoring on how any physical/virtual object or system performs (or behaves) during its entire course of run within the specified environment. NDT tries to monitor the system's or object's functioning status and probable future changes. Using such a dynamic NDT-based system, a network administrator can monitor real-time changes and determine whether any issues or dynamic adaptations are required. Using such a system, a network operator can anticipate how a network will behave in the future, which aids in efficient planning and decision-making competency. As a result, SSM-based systems often allow for well-timed modifications of the present statuses of network elements (or physical objects), guaranteeing that the network can provide high QoS.

In [145], the authors presented a novel mechanism for modeling Computer Numerical Control (CNC) machines tools employing NDT. This model assists in supervising and sustaining these machine tools judiciously. The proposed strategy intends to make it easier to keep CNC machine tools in excellent operating order by anticipating and correcting complications before they become detrimental. However, the model did not consider some physical characteristics such as cutting force, heat, and the pulsation effect, all of which might influence how CNC machine tools perform. Similarly, the authors [241] developed the DT of the smart CNC machine tool to better understand the dynamics of machine tools throughout the machining procedure.

Further, the authors [235] presented a novel DT-based scheme for performing tool monitoring in the context of the milling process. In [132], the authors offered a

TABLE 8. Recent contributions to the use of the DT paradigm in the context of next-generation communication networks.

Ref. (Year)	Network environment	Problem addressed	Contribution/methodology	Target parameter(s)
[259] (2024)	CAVs and RSUs network	The management of CAVs in dynamic wireless networks is inadequate.	The SMDT platform enhances autonomous vehicle supervision over next-gen wireless networks by integrating cloud services and DT capabilities, improving traffic efficiency and safety.	Travel time, blocking probability, and delay.
[297] (2024)	6G	Extremely limited User QoE assessments in 6G network scenarios.	Presents the QoE assessment dataset used to generate the NDT for analyzing offered QoE. Such created NDT assists in analyzing and monitoring the critical user QoE parameters in the case of 6G networks.	Anticipate optimized QoE values.
[301] (2024)	Mobile Networks (edge-cloud systems)	Federated Learning in mobile networks faces challenges due to its need for fast communication and stable connectivity.	The scheme introduces a lightweight Federated Reinforcement Learning framework with a cloud-edge structured DT arrangement for mobile networks, optimizing client node selection and aggregation frequency in the FL procedure.	Resource consumption, training performance, and accuracy.
[135] (2024)	IoT HetNets	Task offloading has a detrimental effect on next-generation network performance because it oversees some important issues.	By optimizing some characteristics, it reduces a crucial factor, namely, the DT job finishing time of all IoT devices.	Task Completion Time
[8] (2023)	Multi-hop IoV (Vehicular networks)	Challenges in mobile networks and IoV include stable connectivity for Federated Learning and high mobility affecting DRL and V2V connectivity.	Describes a DT-assisted DCG mechanism using MADDPG for latency-aware routing optimization in dynamic networks.	Training Loss, Reward, Queue length, Queuing delay, power consumption
[45] (2023)	DT-assisted MEC network environment	There are significant irregularities in MEC environments.	It improves collaboration among complex devices in an IoT system. This work formulated the problem to maximize power efficiency while maintaining a suitable workload balance among edge servers.	Power consumption, server, and user queue length.
[95] (2023)	6G HetNets	In 6G HetNets, mobile users must dynamically associate with base stations to meet QoS requirements.	Presents a mobile user association design that matches base station resources with user load, optimizing the DT model-building process through data resampling and principal component analysis.	Network-wide QoS satisfaction.
[84] (2022)	HCN	FL performance is affected by network congestion, low device computation, and high latency (especially over long distances to cloud servers).	The framework uses a DT network model and edge servers to address user device issues in FL.	System cost of the user device (power consumption).
[172] (2022)	6G Edge Network	The concerns of security and privacy (data) due to the employment of ML and AI schemes are highlighted using blockchaining and transfer learning schemes.	It optimizes critical 6G network metrics by leveraging ML and AI practices. It incorporates transfer learning and blockchain to sustain privacy and security during network expansion and resurgence.	System time cost
[120] (2022)	UAV-assisted MEC network	Given the significant user mobility and irregularity in MEC environments issues.	The framework models a DT-enabled UAV-assisted MEC network, optimizing energy, user association, UAV trajectory, and computation with power and latency constraints.	System power consumption.
[141] (2021)	6G Edge Network	DT deployment in edge networks faces challenges in placement, maintenance, and adapting to dynamic factors.	The framework incorporates DT into the MEC environment, modeling edge network systems and formulating an edge association problem. It solves this using DRL for DT placement and transfer learning for DT migration.	System time cost
[22] (2021)	Industrial IoT-Edge Network	It can be challenging to oversee interoperability across protocols created by different manufacturing tools.	This framework facilitates the generalization of collaborations with sophisticated heterogeneous industrial gadgets by effectively performing dynamic regulation of the use of network resources and improving user QoS in the specified networks.	Processing overhead and communication delay
[265] (2021)	V2X + LTE	Motion control and planning for CAV networks assume optimal V2X communication, which is impractical in real-time wireless systems.	A DT-assisted CAV architecture for signal-free intersections includes FIFO slot reservation, vehicle positioning control, and motion prediction for V2X and LTE communication.	Power consumption and travel time
[48] (2020)	IIoT systems	The dynamic issues in IIoT systems, make it difficult for the network to be efficient.	The approach uses NDT for modeling STAs and network architecture in IIoT systems, optimizing computation offloading and resource allocation to reduce power usage.	Power consumption and system cost
[139] [142] (2020)	Industrial IoT-Edge Network	Integrating edge devices with ML-based schemes is difficult.	Work [139] developed a DT-assisted edge network system using FL for local model re-training on edge devices. Work [142] enhanced this with blockchain technology for better communication security.	Communication competence, power consumption, and model accuracy.
[143] (2020)	6G Edge Network	High latency and reliability are key challenges in edge computing, especially for FL in IIoT.	The scheme maps IoT devices to DTs in edge servers, improving AI performance and reducing the impact of inconsistent wireless communication.	Learning latency and learning convergence.
[228] (2020)	Heterogeneous IIoT system	Traditional methods fail to account for dynamic industrial systems and the need for adaptive FL design.	The scheme uses a DT design to simulate and map industrial devices' behavior in a virtual environment, incorporating trusted-based aggregation in an FL.	Learning convergence, accuracy, and power saving.
[229] (2020)	6G Edge Network	MEC offloading in 6G networks struggles with user mobility and unpredictable behavior.	The approach uses a DITEN scenario, where DTs assess edge server states and provide DRL agent training data, illustrating the complex communication and interactions within the MEC system.	Convergence performance, latency, system cost, task failure
[299] (2020)	Software-defined vehicular networks	SDVN design in vehicle networks is robust and reliable, leveraging computing resources to learn and apply networking strategies based on the surrounding environment.	The work uses SDVNs to expand computation resources in vehicular networks and advocates for a DT-based virtual smart network space to iteratively improve networking schemes adaptively.	Packet delivery ratio and delay.

complete examination of these schemes. After doing rigorous assessments, they emphasized the fact that forecasting and correcting time-varying errors is a research problem and little work has been done to address this issue utilizing the DT paradigm. The authors [132] also provided a novel model that successfully manages the difficulty of time-varying errors (particularly owing to heat effects) while working with CNC machine tools. By employing this model, the system is able to anticipate and compensate for these issues using DT. In another work [131], the authors proposed a system that combines DT with data-driven design to forecast the quality of complicated products during the die-casting method and after machining. They employed an approach of data pre-processing (cleaning and organizing data) and an ML technique called XGBoost to make these real-time quality estimates. They developed a novel DL method, known as a single-shot refinement neural network, to detect minor faults in aluminum castings, particularly when there is a lot of background noise or interference. This method is effective for detecting minor flaws that may be difficult to perceive. Another DT-based approach [113] uses the Particle Swarm Optimization (PSO) technique to achieve online thermal impedance monitoring. This DT-based solution adequately addresses the issue of modeling how heat affects distinct chips that interact closely by utilizing PSO.

C. ML-DL IN NDT FOR RESOURCE ALLOCATION

Resource allocation using a machine and deep learning-enabled NDT is another major aspect that has drawn the attention of many researchers. With DL and ML models, DT technologies allow the system to optimize resource utilization in an efficient way by analyzing vast amounts of data and providing optimal resource allocation decisions. They also enable the system to simulate and monitor the condition of constrained resources in real-time.

System latency is a crucial indicator of a system's overall performance. Rapid information processing and utilization are essential for providing effective services, which in turn improve user experience, guarantee system stability and dependability, and satisfy real-time service requirements. An NDT with machine and deep learning capabilities reduces system latency with these aspects.

To increase the flexibility of a network, network slicing technologies can split a single physical network into several separate virtual network slices. In [121], a dynamic cooperative slicing framework (based on NDT) was proposed to decrease service delays in delay-sensitive services. Every slice has access to its own network resources and is capable of offering stand-alone services. The authors included such network slicing for wireless access into heterogeneous networks. The purpose of this integration was to maximize the reusability of network resources. Moreover, the NDT is used to continuously monitor the condition of every mobile edge server and the entire wireless network. The authors also used MEC servers to supply computational resources for delay-sensitive network slicing, which required low latency and queue stability. The scheme maximized the long-term utility of the network slicing operator by creating a joint optimization problem of network resources, computing resources, and cooperation ratio and using the Deep Deterministic Policy Gradient (DDPG) algorithm with a changeable action space to solve it.

In [225], a Virtual Reality-embedded DT was used to facilitate the operators to see the dynamics of data streams and the internal function of IIoT devices. To enhance the QoS of the VR-DT service (regarding service latency and transaction throughput), the authors presented a blockchain-based distributed resource allocation scheme. This technique models a collaborative optimization of subframe configuration, channel allocation, computational capacity reservation, and block size adaptation as a mixed integer non-linear programming problem. Then, to address the QoS optimization issue, they suggested a fully distributed multi-agent composite action critical actuator algorithm.

Again, in the domain of resource allocation, lowering energy consumption requires less resource usability and communicating less frequently to minimize the system latency. Consequently, it is critical to use NDTs to consider various aspects of dynamic systems, such as network conditions, user demands, and the amount of available resources, in order to determine the best balance between delay and energy consumption.

By using FL to create DTs of physical devices within IoT networks, Liu et al. [133] were able to handle the energy consumption that arises from VNF migration and service function chain in the effective provision of VNF to IoT devices. This allowed automated management and allocation of VNF. They proposed an FL-based bidirectional gated recurrent unit algorithm to predict the dynamically shifting needs for network resources precisely. In this work, the researchers utilized a distributed proximal policy optimization-based DRL method to guide VNF migration decisions by continually monitoring the state of IoT network servers. This strategy attempted to decrease the network's total power consumption and minimize the frequency of VNF migrations.

D. ML-DL IN NDT FOR TASK OFFLOADING

In the era of IoT, we have a vast amount of mobile or handheld devices (e.g., smartphones, wearables, and IoT sensors). As a result, an increased number of computing tasks is generated by these devices, that frequently manage user interactions, real-time monitoring, and data collection. A lot of these devices have limited memory, processing power, and battery life. As a result, carrying out demanding operations locally may burden these resources, which could lower user QoE and overall performance. To this direction offloading and workload scheduling have become feasible in augmenting computing resource utilization competency, and thus, they reduce overall system power consumption.

MEC reduces latency between mobile edge servers and mobile users by bringing cloud resources like storage and processing power to the network's edge. Many MEC-based schemes focus on offloading schemes to trade-off between latency and energy expenditure. A single mobile edge server is deficient in managing multiple workloads concurrently due to deployed smart and complex applications. Here, the notion of edge cooperation and collaboration addresses the restrictions of individual mobile edge servers by supporting a distributed and cooperative attempt at task processing. This method leverages the combined resources of multiple devices to improve performance and reduce power and time overhead, leading to a more applicable and receptive edge computing environment. At the same time, this kind of edge collaboration employs idle network resources by processing workloads in a distributed and cooperative manner [134]. The notion of task offloading moves complicated computing workloads from nearby devices to remote servers or cloud infrastructure for effective execution. As a result, the processing burden on local devices is decreased, and the system performance is improved.

Recently, the application of AI to address several MEC-related problems, including workload offloading and resource allocation [35], [127] constitutes a potential research domain. The authors [134] emphasized that the aforementioned schemes [35], [127] did not take into account the practicality of carrying out intricate AI-based processes on mobile user devices with severe resource constraints. When realizing complex AI-based procedures (typically during neural network training), massive volumes of data are needed. However, the highly constrained storage capacity of mobile user devices makes it impossible for efficient training, resulting in the prediction of imprecise results. Besides, the extremely limited computational resources further limit the usefulness of such devices for network training. Lately, several authors used NDTs in managing issues like intelligently selecting mobile edge servers and intricate AI-based offloading procedures. Considering the benefits of NDT, we consider that NDTs can be integrated with MEC to construct a novel Digital Twin Edge Network (DTEN) [134].

In particular, an NDT comprises DT-based models of edge servers and mobile devices together with a learning model. Under such circumstances, an NDT controls and optimizes the distribution of computing resources among mobile devices' and edge servers' networks. The DT model of mobile devices captures and exhibits each device's real-time status, performance, and resource availability. On the other hand, the DT model of edge servers assists in replicating the state and performance of edge servers. Moreover, an NDT supervises and manages the global state through essential real-time information exchange amongst DT models, maximizing the use of accessible core-constrained resources. These DT models continuously exchange real-time information on their present statuses and the availability of resources, assisting NDT in keeping an updated perspective of the state of the global system. An NDT can decide how well to distribute resources by examining this shared data. This entails figuring out which jobs (given their requirements, their current load, and their capabilities) should be processed on particular servers or devices. However, the learning model uses historical data, real-time inputs, and simulated scenarios to determine the best strategies for offloading tasks. It also determines where workloads should be offloaded by evaluating incoming requests and resource availability. This aids in load balancing and keeps no single server or device from acting as a bottleneck. In addition, an NDT also assists in optimizing these offloading choices via simulated training ultimately to guarantee successful workload completion in extreme constraint circumstances [187]. In general, an NDT aids in training and improving the learning model and related intricate procedures by using simulated settings. This procedure involves simulating and executing numerous scenarios and evaluating results to optimize decision-making. Consequently, through these extensive simulations, the system can anticipate the influence of different choices and adapt its procedures accordingly. When decisions are made, they are shared and broadcasted via edge servers to offload decision implementations in real-time. This guarantees that the optimized plan is followed during the task offloading, resulting in effective execution. In [187], the authors emphasized how combining ML and DL enables an NDT to continuously learn from data and modify its intricate processes on the fly. Typically, it can help in anticipating future probable states, identifying potential issues before they commence, and modifying procedures in real-time. This integration aids NDT in evaluating, anticipating, and regularly assessing the real-time state of the system and the corresponding most current resource availability status [187]. By controlling and streamlining the task distribution process, this model can generally reduce execution time and maximize resource utilization. This further contributes to faster processing and less idle time for devices and servers. Furthermore, regarding effective resource allocation, it also significantly impacts power consumption. In particular, power consumption is decreased by not overloading servers and devices and ensuring that tasks are finished with the best resources available. These NDT models offer mechanisms for the optimal use of computing resources, shorten the time it takes to complete a task, increase computational proficiency, and use less energy. The system considers multiple factors such as task priority, execution time, and resource constraints. This holistic methodology confirms that workloads are managed most efficiently, balancing performance with resource availability.

Many computationally demanding workloads (produced by IoT-based devices) cannot be easily executed rapidly and thoroughly in dynamic contexts when computing resources are limited. The solution to this problem is MEC, which provides support for dynamically offloading resource-intensive tasks to the edge servers. Indeed, many offloading strategies over MEC were provided in the literature. In these strategies, the most significant issue that must be considered is how to precisely determine the system's running status and resource availability. The authors in [134] emphasized the importance of applying AI over MEC to increase the reliability of the MEC system. However, implementing AI-based complicated processes in these dynamic settings is not easy and straightforward. The authors [134] highlighted a key requirement (viz. massive amounts of data are required to realize complicated AI-based operations) which must be addressed usually during neural network training. However, the severely limited storage capacity of mobile devices prevents effective training, which leads to imprecise results prediction. The low processing resources of these devices further restrict their use for network training. Subsequently, the authors [134] presented a DT-based workload offloading scheme based on edge cooperation, which employs the usage of mobile edge server selection. The proposed method suggests choosing and employing a mobile edge server that has effective communication link characteristics as the edge collaboration node, utilizing integrating potent blockchain technology with DT. The blockchain concept and channel state information are used to facilitate effective mobile edge server selection decision-making. Also, the authors in [134] modeled the concept of workload offloading of mobile users as a Markov Decision Process (MDP) and subsequently formulated an optimization model. This model aims to optimize a utilized function that is represented in terms of energy consumption and latency.

To investigate the task offloading problem in DT-enabled MEC systems, the authors in [45] enhanced the collaboration among sophisticated devices in an IoT system. As a result, they formulated the problem to maximize power efficiency while maintaining an appropriate workload balance among edge servers. The proposed offloading technique can identify competent edge servers and determine whether an IoT device should offload a task or not. Then, the authors modeled the workload offloading as an MDP problem and devised an efficient DRL-based energy-efficient task offloading technique to address it.

To make task-offloading decisions and associated methods dynamic, DRL has recently received a lot of attention to assess and select choices where combined utilization needs to be optimized. Among the available DRL techniques, the Deep Q-Network (DQN) has specifically gained a lot of interest from researchers. DQN is commonly considered a specialized variation of Q-Learning. This learning model typically employs the temporal-difference learning from RL and function approximation from DL [280]. In fact, temporal-difference learning from RL means adjusting its Q-value estimations based on the difference between the expected value and the actual reward, gradually altering its policy over time. On the other hand, the function approximation from DL means- since conventional Q-learning scheme employs a Q-table to embrace Q-values for every action-state pair, which can be highly impractical, typically dealing with environments having vast state or action spaces. Instead, DQN approximates the Q-values using deep neural networks, allowing it to deal with considerably larger and more complicated datasets. Traditionally, Q-learning has been used to address various traditional network-related issues, such as power efficiency [216] and routing [226]. However, much more complex versions are now commonly used to manage multidimensional complex optimization problems in specialized and advanced heterogeneous network settings. Recently, Li et al. [120] focused on DT-assisted workload offloading, specifically for Unmanned Areal Vehicle (UAV)-based edge systems, where the Double DQN (DDQN) model is examined to decrease the total system power consumption. Liu et al. [136] presented another effort on the power-efficient task offloading problem, taking into account multi-agent collaboration, typically for the case of aerial edge systems.

From the above discussion, it is observed that merging ML and DL advanced methods with NDT can provide significant answers to difficult workload offloading problems, particularly in multiple MEC-assisted scenarios. In addition, NDT aids in understanding how physical entities (e.g., devices, sensors, etc.) interact and what their features are. ML and DL algorithms can use this data to detect correlations and patterns between different network components. For example, it can estimate which devices will be idle and which will be busy, as well as the optimal times to offload responsibilities. Also, ML and DL algorithms can develop sophisticated service models by analyzing the captured data from the NDT. These models enable the development of effective systems that manage constrained communication and computation resources. The information gathered can help to make more precise and adaptable task-offloading decisions. Also, the dynamic interactions between AI-based procedures and NDT allow for accurate and real-time management of resources. These dynamic systems can assign workloads to underutilized or idle resources by repetitively observing the network's digital counterpart in real-time. This helps to optimize resource utilization and ensure that tasks are performed by the most appropriate device or server at any given time [187]. Undoubtedly, real-time data interpretation through NDT helps with precise workload offloading decisions. This leads to better compute latency performance, especially for edge devices, and faster response and processing times in the end, and finally elevates user QoE and QoS. This is essential in latency-sensitive applications that must analyze data quickly and immediately. Such applications are real-time data analytics or responsive control systems. Ultimately, the specific applications and continuous improvement and enhancement of ML-DL processes inside NDN significantly boost the intelligence of workload offloading decision-making processes while concurrently reducing the edge device power consumption.

E. ML-DL IN NDT FOR MODEL OPTIMIZATION

This subsection presents some research efforts on DL and ML-enabled NDTs for model optimization. Model

TABLE 9. Integration of NDT with ML.

Reason for integration	Problem addressed	NDT applications
Anomaly handling	• Anomaly detection in networks • Fault detection, and maintenance in networks • Intrusion detection • Anomaly monitoring	• Applications based on networks using NFV • Industrial IoT applications • Networking applications
System state monitoring	• How a network or network element will behave in the future?	• Applications that provide high QoS. • Applications that control the DTs of smart CNC machine tools.
Resource allocation	• To optimize resource utilization. • To improve QoS. • To decrease latency	• Monitoring the internal functionality of IIoT devices. • Applications supported by delay-sensitive services. • MEC applications based on network slicing. • Applications that handle NFV migration to IoT devices. • Automated management and allocation of NFV.
Task offloading	• Workload offloading	• MEC-based applications for offloading schemes. • Applications that continuously learn from IoT data and their intricate processes on the fly.
Model optimization	• To enhance predictive capabilities and decision-making processes.	• Applications that improve the training process of DT models. • Applications (of ML-based IoT platforms) that optimize energy and production operations. • Applications that optimize plant operation. • Medical applications that predict health outcomes, personalize treatment plans and enhance patient care.

optimization has many aspects such as automation, accuracy, efficiency, extensibility, and self-iterative updating of the model-building process. To overcome obstacles in the management of a large-scale network and the deficiency of real labels in the early stages of ML-based IoT platform product design, the authors in [265] introduced various approaches like the DT-assisted IoT platform design framework and two-level hierarchical learning algorithms, that improved the predictive accuracy of DT. In addition, a trust-based aggregation model was presented in [228] to improve DT learning performance, particularly in the presence of resource limitations. This model optimized the DT model's learning accuracy and convergence speed by utilizing DQN and Lyapunov dynamic deficit queue techniques to adjust the aggregation frequency.

Many research endeavors are focused on improving the DT models' accuracy through improved interactions with real-world objects. During the interaction process, the physical entity can carry out optimized operations based on the DT model's feedback information, and the optimized physical entity can also help the DT model get even better. The physical entity and the model can continuously improve thanks to this mutual interaction. As an illustration, [36] used VPepper and ML agents to create smooth transitions between the real and digital environments inside the DT system. The robust deployment and varied applications of DT are made possible by these interactive capabilities with robots and smart environments, which also open the door for DT's scalability in a range of scenarios. Improving the training efficiency of NDT models is a critical challenge to deploy these models reliably and effectively in real-world production environments. Some researchers tried to reduce the substantial communication overhead caused by end-to-end delays to speed up the training process. For example, [96] proposed a concurrent end-to-end synchronization and multi-attribute data resampling approach that allowed DT to avoid the effects of sampling rate heterogeneity and distributed sensing process non-synchronization on DT model construction. The proposed strategy uses a penalty regression technique based on Lasso to implement an attribute selection mechanism and sampling rate adjustment driven by feedback. Additionally, the authors used edge devices' computational power to effectively manage the huge amount of data generated by IoT devices, improving network reliability overall. With this all-encompassing method, the DT construction is accomplished with efficiency and accuracy. Furthermore, to maximize the training efficiency of learning models in industrial vision sensing and drive systems, Xu et al. [279] proposed a time-lapse safety perception DT model based on vision perception and drive. This model significantly increases the learning model's training speed by utilizing transfer learning and cross-domain driving based on Q-learning to enhance perceptual accuracy, efficiency, and driving performance, respectively. In [95], a virtual network topology-based DT mobile network was proposed to address the imbalance in the spatiotemporal load of multi-layer base stations during user association. Data resampling and outdated model discovery (based on principal component analysis) are used to optimize the accurate and efficient DT model construction process.

Integrating NDT with ML allows enhanced predictive capabilities and decision-making processes in various industrial domains. The collaboration between DTs, big data, and ML drives significant advancements across various industries. By tying together the power of data and advanced analytics, organizations can create more accurate, dynamic, and intelligent DTs that offer unprecedented insights and optimization opportunities. Undoubtedly, DTs now have a more significant role in transforming industries and determining the future of digital transformation.

In manufacturing industries, supervised learning techniques can be used for work condition estimation, operational optimization, fault prediction, and detection of the equipment based on the collected datasets [140], [180]. To deal with the requirements of the variable operating environments of electromechanical products and the difficulty in acquiring operational data, the authors [140] introduced an operation simulation system and an application framework for electromechanical products powered by a DT. By enhancing the product DT model and creating a virtual operating environment, the system accurately simulates the product's real operating conditions. Authors in [180] introduced a DT architecture designed for energy optimization in manufacturing systems. This architecture focuses on a 'what-if' simulation model, and the effectiveness of this model is demonstrated using a case study involving the efficient energy management of autonomous guided vehicles in a battery pack assembly line. In smart factories, NDTs paired with ML can enhance production operations, forecast equipment breakdowns, and minimize downtime by analyzing real-time data from machinery. Siemens is a pioneer in developing DTs in manufacturing [220]. Their Digital Twin of Production allows for real-time enhancement and monitoring of production operations. By incorporating big data and ML, Siemens can simulate various production scenarios, predict potential bottlenecks, and optimize the manufacturing process to improve competence and reduce costs.

In the healthcare sector, Philips has deployed a DT of patients, known as the "Digital Twin of Care" [178]. This system leverages Big Data from medical records, wearable devices, and other resources to build a complete digital representation of a patient. ML algorithms analyze this data to predict health outcomes, personalize treatment plans, and enhance patient care. Similarly, the authors in [114] introduced a DT framework architecture for health and well-being based on the ISO/IEEE 11073 standard [18]. The framework covers the continuous loop process of gathering data from personal health devices, processing the data, and giving the user response/feedback. It offers a solution for integrating not only X73-compliant devices but also non-compliant health devices by connecting them through an X73 wrapper module. Additionally, they implemented a configurable X73 mobile application, which is compatible with any X73-compliant device. They also designed a framework and accomplished an experiment to demonstrate the DT's effectiveness.

General Electric (GE) [240] has deployed DTs for power plants, enabling real-time supervising and optimization of plant operations. By incorporating big data from sensors and IoT devices with ML algorithms, GE's Digital Twin can forecast equipment breakdowns, enhance energy production, and minimize operational costs. This deployment has proven particularly effective in improving the reliability and efficiency of power generation.

XI. NDT AND NETWORK TRAFFIC ENGINEERING

So far, the survey has covered a wide range of DT and NDT features and applications. However, one of the important aspects of computer networking, called traffic engineering, and how NDT can be used to improve this aspect has not been investigated. In general, traffic engineering consists of two main pillars: traffic measurement and traffic management. The former is responsible for monitoring and analyzing real-time traffic, and its main goal is to provide the necessary prerequisites for traffic management. On the other hand, traffic management includes all traffic routing and steering fundamentals, such as load balancing, QoS guarantee, energy-saving, and traffic management for the hybrid IP/SDN. Both components, working hand in hand, should ensure that the user experience is high and particular QoS is met [6].

By incorporating NDT, traffic engineering can benefit from a deeper understanding of network behavior through real-time, high-fidelity simulations of traffic flow and performance metrics. This allows network operators to predict potential bottlenecks, test various routing strategies, and proactively address challenges before they impact the user experience. In addition, NDT can enable more dynamic and adaptive traffic management approaches to providing a virtual environment to evaluate changes in load balancing, energy efficiency measures, and QoS configurations. This integration not only enhances decision-making processes but also reduces the risks associated with implementing changes in live networks, making traffic engineering more robust and future-proof [285].

Sophisticated traffic engineering mechanisms are indeed essential for routing traffic through the network to avoid some of the common problems that can occur, such as congestion, high latency, and throughput degradation [6]. However, due to the complexity of networks, it is not straightforward to know the exact outcomes of the routing schemes prior to implementation, and sometimes the desired routings lead to adverse impacts on networks' functionality and delivered QoS, where the role of NDT can be emphasized in replicating the network for the experiments to be performed within the framework [10].

Building on this foundation, recent advancements in NDT technology have demonstrated their potential to revolutionize traffic engineering by offering predictive and simulation-based capabilities. For instance, NDTs enable the creation of real-time, high-fidelity virtual replicas of networks, allowing operators to test and evaluate various traffic routing strategies without disrupting the live network. Studies have shown that integrating NDTs with data-driven frameworks can enhance traffic prediction accuracy and improve decision-making processes by leveraging advanced algorithms, such as graph neural networks and machine learning models [165], [285].

Additionally, these systems facilitate a more efficient allocation of resources by simulating traffic flow under different conditions, enabling proactive responses to potential congestion or failures. As a result, NDTs not only improve traffic measurement and management processes but also pave the way for more adaptive, resilient, and scalable network architectures [219].

Generally, traffic engineering can be divided into two main categories: IP- and Multiprotocol Label Switching network (MPLS)-based ones. The main approach for the former is to optimize the IP routing mechanisms through improving load balancing in multi-path networks [82]. Approaches that rely on the IP-based mechanism have two major shortcomings: 1) Since most algorithms rely on Open Shortest Path First (OSPF) [162] weights, they inherit its inability to use the links sufficiently and reach sub-optimal load balancing in particular conditions. 2) When the characteristics of a link change, a link fails, or a topology change occurs, it takes some time for the weights to converge to the optimal values. Both drawbacks can lead to issues such as congestion, packet loss, or high latencies in the network, as a result, degrading the performance of the network [6].

To address the flaws of the IP-based traffic engineering approaches, it was proposed to route data using MPLS labels that can replace IP headers [13]. MPLS could indeed bring some advantages by relying on its predetermined labels for routing instead of using the conventional source/destination IP mechanism. However, its implementation is complex, imposes high-performance overhead, and suffers from difficulties in bursty traffic. To address the issues in both approaches, many novel mechanisms have been proposed that could refine the traffic engineering functionality to some extent. Nevertheless, individuals ran into various flaws and drawbacks [6].

Another way to enhance the routing was to go beyond the traditional routing schemes and separate the data and control planes by using SDN [47], [124]. The core idea behind SDN, which was proposed by Stanford University researchers, is to move the routing decisions to a centralized node that has insight into the entire network. This way, administrators can program and control forwarding decisions, which improves routing and also allows for a novel approach implementation. On the one hand, SDN could provide many benefits, such as centralizing the network information in the controller. As a result, the controller can view the entire network, store the topology, be aware of network changes, and more. In addition, SDN could hone programmability and openness, as the SDN switches can communicate with the controller in a unified way [42], [63].

On the other hand, SDN also suffers from several drawbacks. One of the most important issues is the compatibility of SDN with the traditional routing algorithm, which can be problematic [4], [239], thus creating a hybrid IP/SDN routing might be challenging [109]. Furthermore, concentrating the control process in one or a few distributed centralized nodes requires some effort to manage and maintain, in a way that should prevent failure due to controller irresponsibility and avert the single-point-of-failure problem [77].

To further mitigate these challenges, integrating emerging technologies like NDTs with SDN-based traffic engineering offers a promising direction. As NDTs provide real-time, virtual replicas of networks that can simulate and predict network behavior, enabling more efficient routing and resource allocation. By combining the centralized control of SDN with the predictive and simulation capabilities of NDTs, network administrators can proactively address issues such as link failures, congestion, or topology changes. This integration also facilitates the testing and deployment of hybrid IP/SDN routing mechanisms, reducing compatibility issues and improving overall network resilience. However, implementing such advanced systems requires addressing new challenges, including the computational demands of maintaining real-time digital twins and ensuring the scalability of the combined framework in large, dynamic networks. These considerations highlight the need for continued research and development to fully realize the potential of NDT-enhanced SDN traffic engineering [79], [263], [267].

Besides the issues mentioned, there is another important hurdle in the design of a new traffic engineering mechanism, which is the implementation part. Unless the algorithm is not analyzed in a real network, the performance cannot be fully reliable. However, the practical implementations have their own difficulties, and if there is a problem with the algorithm, they may interrupt the services or degrade the QoS for the applications. This issue can be more exacerbated during the peak hours when more users are using the network, and service disruptions can impair the user experience of many clients. Moreover, as the network becomes more complex, the mechanisms can be difficult to implement, debug, evaluate, and optimize [195], [282]. To address this issue, one way is to use simulation and emulation tools such as NS-3 [198] or Mininet [102]. However, as was mentioned, these tools suffer from accuracy, speed, and the lack of bi-directional communication with the physical network. As a result, many researchers have tried to use NDT as an enabler to improve traffic routing in computer networking.

To overcome the challenges of implementing new traffic engineering algorithms in real-world networks, researchers have increasingly turned to NDTs as a solution. By replicating real-time network behavior, NDTs enable the evaluation and optimization of traffic engineering mechanisms in a controlled environment. These digital replicas offer significant advantages over traditional simulation tools, such as NS-3 and Mininet, which suffer from limitations in accuracy, speed, and the lack of direct interaction with the physical network [80], [263].

With NDTs, it becomes possible to simulate traffic flows, predict potential disruptions, and test new algorithms under a variety of network conditions, thus reducing the likelihood of performance degradation during peak usage times. Moreover, NDTs can facilitate bi-directional communication with the physical network, enabling real-time adjustments and ensuring that the traffic engineering mechanisms are optimized for both current and future demands. This ability to test in a realistic yet risk-free virtual environment has made NDTs a key enabler in advancing traffic routing technologies [80], [267].

The first step on this path is to map the topology and see how different traffic patterns can be reflected in the NDT, which was done by creating a new tool called RouteNet-Erlang using GNN in [66]. This tool attempts to replicate the network by representing the links, queues, and source-destination flows. The main goal of the tool was to use data-driven approaches to model complex traffic models. There was another attempt to enhance the weight assignment of the OSFP by using NDT in [291]. This work relied on QoS metrics, mainly end-to-end delay, to achieve its goal and was successful in reducing the end-to-end delay. The results showed that this method can reduce the traffic delay of twinned networks. Nevertheless, the method relies on a limited number of weights, which makes it doubtful whether it can be generalized to more complex networks or not.

TwinNet [65] was another NDT-based attempt at modeling the network, routing configurations, and input traffic matrices. This tool also relies on GNN to figure out the relationship between the different components of the network, such as scheduling algorithms, queue sizes, and the network topology. TwinNet strives to attain its goals by accurately estimating the Service Level Agreements (SLA) in such a way that it not only performs well on the trained dataset, but also generalizes to the unseen data. Evaluation results of the model have shown that TwinNet can estimate end-to-end path delays in various what-if traffic scenarios over a wide variety of topologies and routing configurations.

Similarly, NDT-based solutions have been explored to enhance existing routing algorithms, such as OSPF, by improving the weight assignment process. In the study by Zalat et al. [291], the authors focused on incorporating QoS metrics, particularly end-to-end delay, into the NDT framework. The method demonstrated a reduction in end-to-end delay, which is critical for maintaining high performance in time-sensitive applications. However, the approach was constrained by its reliance on a limited number of weights, raising concerns about its scalability and generalizability to more complex network topologies.

An intelligent SDN-based routing, even for vehicular networks, has also been proposed in [298], called intELligent dIgital Twin hiErarchical (ELITE), which aims to improve routing in Software-Defined Vehicular Network (SDVN). ELITE's main attempt is to rely on corresponding physical objects within the digital twin to create virtual instances in a way that can overcome the flaws of routing policies in SDVNs; flaws such as frequent topological changes, complex service requests, and long model training time. The proposed hierarchical routing scheme could achieve enhancements in terms of packet delivery ratio, end-to-end delay, and communication overhead compared to its counterparts.

The combination of SDN routing and NDT also over optical networks has been proposed in [252]. The main leverage of the proposal is to use the SDN controller as an enabler of NDT deployment over an optical network to have improved SDN-based routing. In this way, the physical optical network can be virtually represented in the NDT to test the implementation schemes before the deployment. In this way, the use of NDT by a cloud-native SDN controller can enhance the maneuverability of the administration in testing, debugging, and finalizing novel approaches before the implementation, which can save costs and improve the user experience.

The use of Programming Protocol-independent Packet Processors (P4) switches to create distributed digital twin networks was investigated in [116]. P4, based on providing a framework for high-level programming language in order to process packets, can be a useful tool in creating NDT for routing as it functions in concomitance with SDN [29]. The authors attempted to use network virtualization to improve the scalability of the number of twins and locations. The main goal of the work was to provide a framework for having multiple virtual networks, here twins, that can coexist with the physical network in such a way that twins can be distributed throughout the physical network.

Another important aspect of traffic engineering is the prediction of future traffic patterns. This traffic prediction can leverage historical data in a way that can train a model for creating proactive methods for measuring traffic. Many network stakeholders, such as network resource planning, allocation, and management, can benefit from such sophisticated traffic predictions. An attempt towards this goal for predicting the background traffic matrix was made in [115] for LANs. The main contribution of the research was to predict the traffic in a synchronized manner with an NDT model, which helped to enhance the accuracy. The exploited model for mapping the LAN network traffic was convolutional long short-term memory (ConvLSTM), which could achieve high accuracy during the evaluations.

Since traffic engineering and sufficient Transportation Big Data (TBD) can play a major role in the performance of various use case scenarios, such as Vehicular Ad-Hoc Networks (VANETs), an attempt has been made in [169] to use NDT to improve it. As the characteristics of nodes in VANETs are constantly changing, the predictive power and high accuracy of the NDT in TBD can be useful. The main goal of the proposal in [169] was to focus on time-varying traffic flow control, which can be improved by leveraging the predictive feature of the NDT. The mentioned goal could be achieved by combining two machine learning algorithms called DQN and GAN.

As was mentioned in the previous sections, simulation and emulation tools cannot replicate physical networks with high accuracy. One of the appealing ways to increase this accuracy is relying on the NDT capabilities, which was done in [285]. With traffic engineering insight, the authors proposed to emulate flows by utilizing NDT to enhance traffic engineering. The main goal was to keep the mapped model in NDT synchronized with the physical network traffic. This synchronization was the main key in achieving an accurate flow emulation framework. They then used their emulation framework to evaluate it in an application, which showed the enhancements over the achieved delays. However, this model

TABLE 10. NDT in traffic engineering.

Ref.	Summary
RouteNet-Erlang [66]	A tool for mapping the network topology and testing different traffic patterns.
Network routing optimization using Digital Twins [291]	Improving the weight assignment of the OSFP.
TwinNet [65]	Understanding the relationship between the different components of the network, such as scheduling algorithm and queue sizes, the network topology.
ELITE: intelligent SDN-based routing for Vehicular Networks [298]	Relying on corresponding physical objects within the digital twin to create virtual instances to improve routing policies in SDVNs.
SDN routing and NDT over optical networks [252]	Using the SDN controller as an enabler of NDT deployment over optical network to have an improved SDN-based routing.
Using P4 switches to create digital twin distributed networks [116]	Employing network virtualization to improve the scalability for the number of twins and sites.
Traffic prediction [115]	The prediction of background traffic matrix in LAN networks relying on NDT.
Focusing on the big data transportation [169]	combining called DQN and GAN to create an NDT model to improve time-varying traffic flow steering.
Emulation framework for network traffic engineering [285]	Emulate flows by utilizing NDT in order to employ it to enhance traffic engineering.
A model for network traffic prediction relying on NDT [219]	Creating a data set by using OMNeT++ and using it through GNN and RNN to map the network characteristics within NDT.

has a flaw, which is its one-way directional communication as it cannot send control messages to the network.

Another attempt to improve network traffic engineering through traffic prediction was done in [219] with the help of NDT. To do so, a data set was created by using OMNeT++, and then it was passed through GNN to learn the network. Subsequently, RNN was used to be trained on the network performance generated data to map the network behavior. The latter one, which created the functional model of the NDT, was the main tool in predicting the coming traffic in the network. All the processes were performed by using a newly proposed architecture for data flow within the NDT, which could attain acceptable loss and average training time. Table 10 includes a summary of the covered works.

To sum up, traffic engineering is one of the most important aspects of networking that plays a major role in the delivery of a high user experience. However, due to the complexity, dynamic characteristics, and high demand for bandwidth in the networks, it is not straightforward to perform the measurement and management aspects of traffic engineering. NDT, because of its high accuracy, speed, bidirectional communication with the physical network, and ability to predict future patterns, can be used to overcome the existing shortcomings in traffic engineering. NDT can provide a framework that maps the current status of the network and flows, and also helps to make proper decisions by looking at the future patterns.

XII. OPEN RESEARCH CHALLENGES

In this section, we discuss a few unresolved issues in the context of NDT.

A. SECURITY VULNERABILITY

It is a challenging task to guarantee the privacy and security of network elements (e.g., IoT devices, controllers, and routers). Designing an effective authentication mechanism to prevent unauthorized access to servers, DTs, and other network elements is difficult because of their vast scale and distribution. For example, the sharing of information within the NDT may cause security problems. A pair of twins in NDT has a reciprocal feedback relationship. Even though the actual network element would not be readily threatened, an attacker could quickly alter the virtual model or the data it uses to give back information. This type of attack can harm real-time applications such as IoV-oriented applications. So, we must place mechanisms for security reasons that will permit one to understand whether a DT is trustworthy. Notably, security problems may be introduced into NDT by the computationally demanding data analytical tasks handled by edge or cloud servers [9].

B. PRIVACY LEAKAGE

Maintaining data privacy and model privacy in DT is also difficult. For example, to build a virtual model of an IoT device, it is necessary to gather a variety of data about this device in addition to keeping track of and modeling details about the IoT device's environment. Service providers on the cloud or edge servers will handle and evaluate the sensitive data gathered. Driven by profit, these providers typically retain this data indefinitely. They may even share it with other operators without the user's permission, raising the possibility of privacy leaks. Therefore, one of the biggest challenges facing NDT is figuring out how to strike a compromise between protecting sensitive data and maximizing data consumption. Additionally, we notice two problematic cases [105] that could happen: (i) a malicious IoT device transmits error information to mislead the behavior of the edge server for global model training; (ii) a malicious edge server may transmit erroneous global model parameters to mislead the updating of the local models in every IoT device. In both cases, privacy leakage will occur and must be considered when designing the NDT.

C. COST-EFFECTIVE SOLUTIONS

The use of ML, big data, and other technologies is necessary for building an initial complete NDT. It is important to consider the cost of data collection based on extensive sensor deployment. In addition, careful consideration must be given to hardware use, connectivity, computation, storage, and other resources. It is significant to consider how to build an NDT while minimizing overspending [274]. In particular, an NDT needs high-performance information technology infrastructures that can run computationally demanding ML and DL algorithms. High-performance graphics processing units (GPUs) will be required to accomplish this goal. The idea of GPUs as an on-demand service will be crucial in overcoming this obstacle. Masaracchia et al. [151] state that large companies like Amazon and Microsoft are expected to offer distinctive on-demand services that resemble conventional cloud-based apps, removing the barrier of exorbitant prices.

D. REAL-TIME TWO-WAY INTERACTION

Virtual twins use real-time two-way communication to receive data and control IoT devices or network elements. The dynamic nature of the network environment makes it difficult to send such vast amounts of data in real-time. Since the connection is wireless, a low-quality transmission link and thus prolonged service delay could arise from the stochastic nature of the wireless channel. Two-way real-time communication is affected by the NDT's bandwidth, computation load, storage capacity, and energy usage. Both the ongoing model updating and AI prediction need complex computational processes. All things considered, any of them could affect how connected the two-way communication is [274]. Another challenge is connectivity between a DT and its physical twin system, particularly when many sensors must be connected at once and real-time control is required. Indeed, missing IoT data from one or more devices due to connectivity issues could impact the accuracy of data mining and subsequently the system's performance, as sensors and other IoT devices in general serve as sources of data for AI algorithms. Subsequently, it is necessary to develop robust solutions and a high degree of connectivity to prevent software errors and power outages at sensors.

E. SCALABILITY TO REAL NETWORKS

Scalability is an issue for ML models because modern communication networks are frequently bigger than the network settings that produce the training datasets. When used in networks that are significantly larger than those used for training (e.g., 1-2 orders of magnitude larger), NDTs should perform well. Nevertheless, it frequently entails dealing with out-of-distribution quantities (such as higher traffic volumes and link capacity), which could deteriorate the NDT's performance. As a result, developing scalable NDTs is a problem that must be addressed for creating solutions ready for production [11].

F. EFFECTIVE IMPLEMENTATION OF BLOCKCHAIN TECHNOLOGY IN DIGITAL TWINS

Such implementation is impeded by several issues, including scalability, standardization, regulations, data privacy, and interoperability. An effective physical-virtual synchronization is also required to improve interaction between real and virtual objects. Furthermore, to minimize costs and accelerate product delivery, cognitive behavior must be integrated into the blockchain-based DTs environment through the use of federated learning, deep learning, and ML techniques. Integrating blockchain into the NDT is also a hopeful technique in the centralized model training case. In this case, all real things (e.g., IoT devices) send their information to a centralized server for model training. This may cause privacy and security problems during data transmission, and model training. Thanks to blockchain technology, these problems can be resolved [287].

XIII. FUTURE DIRECTIONS

A. CREATING CONSISTENT AND ACCURATE DATA IN REAL-TIME

A communication network infrastructure and its users must be represented with high-fidelity. This is the only method by which a network operator can forecast network traffic overload and design the best possible network configurations to effectively handle it. In particular, high-fidelity representation of all processes running into physical twin counterparts/network elements (e.g., routers, and IoT devices) requires the availability of consistent and accurate data from the network elements in real-time. This will make it possible to take two steps: (1) to optimize the communication network's overall process flow and (2) to identify specific configuration schemes that require extra consideration. For example, in the IoV context, a high-fidelity representation of smart vehicles on the roads can help avoid traffic congestion or collisions. High-fidelity representation is contingent upon the particular use case of DT, while missing data from one of several physical twin entities in context may degrade the entire performance. This implies that data must be continuous, noise-free, and constant when it is transferred between the DT and the physical twin.

B. ACCURATE MOBILITY PREDICTION MODELS

Mobility is a significant challenge for MANET and IoV applications. If a mobile device (connected to a DT) moves outside the coverage area of the twin's access point or base station, it may experience a service interruption. Mobile devices, such as smart vehicles, must have constant, seamless connectivity. Depending on user mobility prediction, handover to a different (possibly nearby) DT object could be one way to resolve the issue. Before the physical system connects to the new DT, we need to create a highly accurate mobility prediction model and effective solutions for both model and learning transfer from one DT to another.

C. SCALABLE APPROACHES FOR WIDE-SCALE ACCESS OF IOT DEVICES

Another important issue to address is the management of the enormous number of IoT devices that are anticipated in 6G networks. Grant-based random access protocols (frequently used in modern networks) may cause significant delays in scheduling or, worse, make it impossible for certain IoT devices to link to the network. To obtain access to the wireless network, each user/IoT device must select a preamble from a pool of orthogonal sequences based on the operation of grant-based random access protocols. The likelihood that two IoT devices will select the same random access sequence is high because these sequences are selected from a limited set. This results in the loss of synchronization between DT and the physical twin as the last one must periodically connect to its corresponding DT replica through a random access procedure. The design of suitable multiple access techniques is currently the most widely used strategy to address this problem. However, new multiple access techniques are not easy to implement because conventional multiple access theory is predicated on long message transmission, and short packets are typically used in massive access [151]. Another direction is starting to use network DTs to do a proper resource allocation aimed at ensuring network access to all of the devices found in a certain area of interest. This process can be scaled up when a large number of devices are introduced. Then, new scalable strategies must be developed and put forth. In addition, AI optimization and compression techniques must be researched to enhance performance in resource-constrained NDT environments. Finally, the use of federated DTs for collaboration and data sharing among distributed NDTs must be investigated [218].

XIV. CONCLUSION

On the one hand, NDT is an evolving technology that is gaining high potential to be used in different areas of computer networking. On the other hand, traffic engineering is one of the most significant pillars of networking and telecommunications, which plays an important role in the delivered user experience and quality of service for various applications. Given these facts, in this survey, we have analyzed NDT, its evolution path, and standardization, with a particular focus on networking, telecommunications, and their prominent pillar, traffic engineering. The survey covers different aspects and algorithms of networking, telecommunications, and traffic engineering; the potential of the NDT deployment for probable improvements in telecommunications; different related networking areas where traffic engineering plays a major role; and where the combination with NDT can bring remarkable benefits. The survey could show that NDT can provide a real-time, highly accurate simulation, testing, and optimization framework, allowing for proactive management and reducing the risks associated with the direct deployment of novel telecommunications and traffic engineering approaches. This improvement can span a wide range of telecommunications areas, including 5G/6G, the Industrial Internet of Things, smart cities, and intelligent transportation. However, it was shown that despite the benefits that NDT can bring to networking and telecommunications, it suffers from some challenges, such as data collection and integration, deployment and development of sophisticated machine learning techniques, the need for real-time processing capabilities, and scalable architecture. In conclusion, NDT can provide a robust framework with high accuracy to be an enabler in order to develop and test novel telecommunications and traffic engineering approaches to be used in a vast range of networking domains.

REFERENCES

[1] H. Ahmadi, A. Nag, Z. Khar, K. Sayrafian, and S. Rahardja, "Networked twins and twins of networks: An overview on the relationship between digital twins and 6G," *IEEE Commun. Standards Mag.*, vol. 5, no. 4, pp. 154–160, Dec. 2021.

[2] W. Ahmed, "Simulation and evaluation of wired and wireless networks with NS_2, NS_3 and OMNET++," Univ. Vaasa, Vaasa, Finland, Tech. Rep., 2015.

[3] G. Aireen, C. E. Mohan, C. H. Pooja, F. T. Pooja, and K. M. Raghuram, "Wireless network simulation and analysis using qualnet," in *Proc. 2nd Int. Conf. Commun. Electron. Syst. (ICCES)*, Oct. 2017, pp. 251–255.

[4] I. F. Akyildiz, A. Lee, P. Wang, M. Luo, and W. Chou, "A roadmap for traffic engineering in SDN-OpenFlow networks," *Comput. Netw.*, vol. 71, pp. 1–30, Oct. 2014, doi: 10.1016/j.comnet.2014.06.002.

[5] J. Li, S. Guo, W. Liang, Q. Chen, Z. Xu, W. Xu, and A. Y. Zomaya, "Digital twin-assisted, SFC-enabled service provisioning in mobile edge computing," *IEEE Trans. Mobile Comput.*, vol. 23, no. 1, pp. 393–408, Jan. 2024, doi: 10.1109/TMC.2022.3227248.

[6] Z. Shu, J. Wan, J. Lin, S. Wang, D. Li, S. Rho, and C. Yang, "Traffic engineering in software-defined networking: Measurement and management," *IEEE Access*, vol. 4, pp. 3246–3256, 2016, doi: 10.1109/ACCESS.2016.2582748.

[7] 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.

[8] M. Z. Alam, K. S. Khan, and A. Jamalipour, "Multiagent best routing in high-mobility digital-twin-driven Internet of Vehicles (IoV)," *IEEE Internet Things J.*, vol. 11, no. 8, pp. 13708–13721, Apr. 2024.

[9] C. Alcaraz and J. Lopez, "Digital twin: A comprehensive survey of security threats," *IEEE Commun. Surveys Tuts.*, vol. 24, no. 3, pp. 1475–1503, 3rd Quart., 2022.

[10] Z. H. Ali, N. El-Rashidy, M. A. Elhosseini, and S. M. Ayyad, "SDN-based reliable emergency message routing schema using digital twins for adjusting beacon transmission in VANET," *J. Netw. Comput. Appl.*, vol. 230, Oct. 2024, Art. no. 103944, doi: 10.1016/j.jnca.2024.103944.

[11] P. Almasan, M. Ferriol-Galmés, J. Paillisse, J. Suárez-Varela, D. Perino, D. López, A. A. P. Perales, P. Harvey, L. Ciavaglia, L. Wong, V. Ram, S. Xiao, X. Shi, X. Cheng, A. Cabellos-Aparicio, and P. Barlet-Ros, "Network digital twin: Context, enabling technologies, and opportunities," *IEEE Commun. Mag.*, vol. 60, no. 11, pp. 22–27, Nov. 2022.

[12] S. Almeaibed, S. Al-Rubaye, A. Tsourdos, and N. P. Avdelidis, "Digital twin analysis to promote safety and security in autonomous vehicles," *IEEE Commun. Standards Mag.*, vol. 5, no. 1, pp. 40–46, Mar. 2021.

[13] G. Armitage, "MPLS: The magic behind the myths [multiprotocol label switching]," *IEEE Commun. Mag.*, vol. 38, no. 1, pp. 124–131, Jan. 2000, doi: 10.1109/35.815462.

[14] P. Arthurs, L. Gillam, P. Krause, N. Wang, K. Halder, and A. Mouzakitis, "A taxonomy and survey of edge cloud computing for intelligent transportation systems and connected vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 7, pp. 6206–6221, Jul. 2022.

[15] (2017). *Modelica—A Unified Object-oriented Language for Systems Modeling Language Specification*. [Online]. Available: http://www.modelica.org

[16] D. Awduche, J. Malcolm, J. Agogbua, M. O'Dell, and J. McManus, *Requirements for Traffic Engineering Over MPLS*, document RFC 2702, RFC Editor, Sep. 1999. [Online]. Available: https://www.rfc-editor.org/rfc/rfc2702

[17] K. Aykurt and W. Kellerer, "Autonomous network management in multi-domain 6G networks based on graph neural networks," in *Proc. IEEE 9th Int. Conf. Netw. Softwarization (NetSoft)*, Jun. 2023, pp. 338–341, doi: 10.1109/NetSoft57336.2023.10175480.

[18] H. F. Badawi, F. Laamarti, and A. El Saddik, "ISO/IEEE 11073 personal health device (X73-PHD) standards compliant systems: A systematic literature review," *IEEE Access*, vol. 7, pp. 3062–3073, 2019.

[19] L. Bao, Q. Wang, and Y. Jiang, "Review of digital twin for intelligent transportation system," in *Proc. Int. Conf. Inf. Control, Electr. Eng. Rail Transit (ICEERT)*, Oct. 2021, pp. 309–315.

[20] Q. Bao, G. Zhao, Y. Yu, S. Dai, and W. Wang, "Ontology-based modeling of part digital twin oriented to assembly," *Proc. Inst. Mech. Eng., B, J. Eng. Manuf.*, vol. 236, nos. 1–2, pp. 16–28, Jan. 2022.

[21] B. R. Barricelli, E. Casiraghi, and D. Fogli, "A survey on digital twin: Definitions, characteristics, applications, and design implications," *IEEE Access*, vol. 7, pp. 167653–167671, 2019, doi: 10.1109/ACCESS.2019.2953499.

[22] P. Bellavista, C. Giannelli, M. Mamei, M. Mendula, and M. Picone, "Application-driven network-aware digital twin management in industrial edge environments," *IEEE Trans. Ind. Informat.*, vol. 17, no. 11, pp. 7791–7801, Nov. 2021.

[23] G. Bernárdez, J. Suárez-Varela, A. López, B. Wu, S. Xiao, X. Cheng, P. Barlet-Ros, and A. Cabellos-Aparicio, "Is machine learning ready for traffic engineering optimization?" in *Proc. IEEE 29th Int. Conf. Netw. Protocols (ICNP)*, Nov. 2021, pp. 1–11.

[24] K. Bonawitz, H. Eichner, W. Grieskamp, D. Huba, A. Ingerman, V. Ivanov, C. Kiddon, J. Koneč ný, S. Mazzocchi, H. B. McMahan, T. V. Overveldt, D. Petrou, D. Ramage, and J. Roselander, "Towards federated learning at scale: System design," in *Proc. Mach. Learn. Syst.*, Jan. 2019, pp. 374–388.

[25] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the Internet of Things," in *Proc. MCC workshop Mobile Cloud Comput.*, Aug. 2012, pp. 13–16.

[26] C. Bormann, A. P. Castellani, and Z. Shelby, "CoAP: An application protocol for billions of tiny internet nodes," *IEEE Internet Comput.*, vol. 16, no. 2, pp. 62–67, Mar. 2012.

[27] R. Rosen, S. Boschert, and A. Sohr, "Next generation digital twin," in *Proc. TMC*, vol. 60, Oct. 2018, pp. 86–96.

[28] S. Boschert and R. Rosen, "Digital twin—The simulation aspect," in *Mechatronic Futures: Challenges and Solutions for Mechatronic Systems and Their Designers*. Berlin, Germany: Springer, 2016, pp. 59–74.

[29] P. Bosshart, D. Daly, G. Gibb, M. Izzard, N. McKeown, J. Rexford, C. Schlesinger, D. Talayco, A. Vahdat, G. Varghese, and D. Walker, "P4: Programming protocol-independent packet processors," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 44, no. 3, pp. 87–95, Jul. 2014.

[30] E. Bottani, A. Cammardella, T. Murino, and S. Vespoli, "From the cyber-physical system to the digital twin: The process development for behaviour modelling of a cyber guided vehicle in M2M logic," in *Proc. 22nd Summer School Francesco TurcoIndustrial Syst. Eng.*, 2017, pp. 1–7.

[31] O. Bousquet and A. Elisseeff, "Stability and generalization," *J. Mach. Learn. Res.*, vol. 2, pp. 499–526, Jan. 2002.

[32] H. Boyes, B. Hallaq, J. Cunningham, and T. Watson, "The industrial Internet of Things (IIoT): An analysis framework," *Comput. Ind.*, vol. 101, pp. 1–12, Oct. 2018.

[33] A. Burkov, *The Hundred-Page Machine Learning Book*. Canada: Andriy Burkov, 2019.

[34] K. Campbell, L. Cruz, B. Flanagan, B. Morelli, B. O'Neil, S. Téral, and J. Watson, "The 5G economy: How 5G will contribute to the global economy," IHS Markit, London, U.K., Tech. Rep., 2019.

[35] Y. Cao, R. Wang, M. Chen, and A. Barnawi, "AI agent in software-defined network: Agent-based network service prediction and wireless resource scheduling optimization," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 5816–5826, Jul. 2020.

[36] L. Cascone, M. Nappi, F. Narducci, and I. Passero, "DTPAAL: Digital twinning pepper and ambient assisted living," *IEEE Trans. Ind. Informat.*, vol. 18, no. 2, pp. 1397–1404, Feb. 2022.

[37] S. Chakrabarty, D. W. Engels, and L. Wood, "Consumer frameworks for smart environments," in *Proc. IEEE 10th Int. Conf. Consum. Electron. (ICCE-Berlin)*, Nov. 2020, pp. 1–5.

[38] X. Chang, "Network simulations with OPNET," in *Proc. Winter Simulation Conf. Bridge Future*, vol. 1, 1999, pp. 307–314.

[39] D. Chen, P. Wawrzynski, and Z. Lv, "Cyber security in smart cities: A review of deep learning-based applications and case studies," *Sustain. Cities Soc.*, vol. 66, Mar. 2021, Art. no. 102655.

[40] J. Chen, C. Yi, S. D. Okegbile, J. Cai, and X. Shen, "Networking architecture and key supporting technologies for human digital twin in personalized healthcare: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 26, no. 1, pp. 706–746, 1st Quart., 2024.

[41] L. Chen, J. Lingys, K. Chen, and F. Liu, "AuTO: Scaling deep reinforcement learning for datacenter-scale automatic traffic optimization," in *Proc. Conf. ACM Special Interest Group Data Commun.*, Aug. 2018, pp. 191–205.

[42] M. Chen, Y. Qian, S. Mao, W. Tang, and X. Yang, "Software-defined mobile networks security," *Mobile Netw. Appl.*, vol. 21, no. 5, pp. 729–743, Jan. 2016, doi: 10.1007/s11036-015-0665-5.

[43] M. Chen, Z. Yang, W. Saad, C. Yin, H. V. Poor, and S. Cui, "A joint learning and communications framework for federated learning over wireless networks," *IEEE Trans. Wireless Commun.*, vol. 20, no. 1, pp. 269–283, Jan. 2021.

[44] X. Chen, E. Kang, S. Shiraishi, V. M. Preciado, and Z. Jiang, "Digital behavioral twins for safe connected cars," in *Proc. 21st ACM/IEEE Int. Conf. Model Driven Eng. Lang. Syst.*, Oct. 2018, pp. 144–153.

[45] Y. Chen, W. Gu, J. Xu, Y. Zhang, and G. Min, "Dynamic task offloading for digital twin-empowered mobile edge computing via deep reinforcement learning," *China Commun.*, vol. 20, no. 11, pp. 164–175, 2023.

[46] C. Cimino, E. Negri, and L. Fumagalli, "Review of digital twin applications in manufacturing," *Comput. Ind.*, vol. 113, Dec. 2019, Art. no. 103130.

[47] J. C. Correa Chica, J. C. Imbachi, and J. F. Botero Vega, "Security in SDN: A comprehensive survey," *J. Netw. Comput. Appl.*, vol. 159, Jun. 2020, Art. no. 102595, doi: 10.1016/j.jnca.2020.102595.

[48] Y. Dai, K. Zhang, S. Maharjan, and Y. Zhang, "Deep reinforcement learning for stochastic computation offloading in digital twin networks," *IEEE Trans. Ind. Informat.*, vol. 17, no. 7, pp. 4968–4977, Jul. 2021.

[49] (2019). *Delmia Global Operations*. Accessed: May 20, 2024. [Online]. Available: https://www.3ds.com/products-services/delmia/

[50] J. Deng, Q. Zheng, G. Liu, J. Bai, K. Tian, C. Sun, Y. Yan, and Y. Liu, "A digital twin approach for self-optimization of mobile networks," in *Proc. IEEE Wireless Commun. Netw. Conf. Workshops (WCNCW)*, Mar. 2021, pp. 1–6.

[51] L. Deren, Y. Wenbo, and S. Zhenfeng, "Smart city based on digital twins," *Comput. Urban Sci.*, vol. 1, no. 1, pp. 1–11, Dec. 2021.

[52] *Digital Twin Market*. Accessed: Aug. 12, 2024. [Online]. Available: https://www.marketsandmarkets.com/Market-Reports/digital-twinmarket-225269522.html?

[53] K. Ding, F. T. S. Chan, X. Zhang, G. Zhou, and F. Zhang, "Defining a digital twin-based cyber-physical production system for autonomous manufacturing in smart shop floors," *Int. J. Prod. Res.*, vol. 57, no. 20, pp. 6315–6334, Oct. 2019.

[54] R. Dong, C. She, W. Hardjawana, Y. Li, and B. Vucetic, "Deep learning for hybrid 5G services in mobile edge computing systems: Learn from a digital twin," *IEEE Trans. Wireless Commun.*, vol. 18, no. 10, pp. 4692–4707, Oct. 2019, doi: 10.1109/TWC.2019.2927312.

[55] J. Downey, D. O'Sullivan, M. Nejmen, S. Bombinski, P. O'Leary, R. Raghavendra, and K. Jemielniak, "Real time monitoring of the CNC process in a production environment—The data collection & analysis phase," *Proc. CIRP*, vol. 41, pp. 920–926, Jan. 2016.

[56] (2021). *Draft New Recommendation ITU-T Y.DTN-REQARCH: 'requirements and Architecture of Digital Twin Network*. Accessed: Jul. 11, 2024. [Online]. Available: https://www.itu.int/md/T17-SG13-210301-TD-WP1-0762

[57] O. E. Marai, T. Taleb, and J. Song, "Roads infrastructure digital twin: A step toward smarter cities realization," *IEEE Netw.*, vol. 35, no. 2, pp. 136–143, Mar. 2021.

[58] T. Erol, A. F. Mendi, and D. Dogan, "The digital twin revolution in healthcare," in *Proc. 4th Int. Symp. Multidisciplinary Stud. Innov. Technol. (ISMSIT)*, Oct. 2020, pp. 1–7.

[59] (2024). *Eve—The Emulated Virtual Environment for Network, Security and Devops*. Accessed: Sep. 11, 2024. [Online]. Available: https://www.eve-ng.net/

[60] F. Tao, J. Cheng, Q. Qi, M. Zhang, H. Zhang, and F. Sui, "Digital twin-driven product design, manufacturing and service with big data," *Int. J. Adv. Manuf. Technol.*, vol. 94, nos. 9–12, pp. 3563–3576, Feb. 2018.

[61] C. Fan, C. Zhang, A. Yahja, and A. Mostafavi, "Disaster city digital twin: A vision for integrating artificial and human intelligence for disaster management," *Int. J. Inf. Manage.*, vol. 56, Feb. 2021, Art. no. 102049.

[62] S. Faye, M. Camelo, J.-S. Sottet, C. Sommer, M. Franke, J. Baudouin, G. Castellanos, R. Decorme, M. P. Fanti, R. Fuladi, G. Kesik, B. Mendes, C. Murphy, S. Parker, S. Pryor, S. M. Senouci, and I. Turcanu, "Integrating network digital twinning into future AI-based 6G systems: The 6G-TWIN vision," in *Proc. Joint Eur. Conf. Netw. Commun. 6G Summit (EuCNC/6G Summit)*, Jun. 2024, pp. 883–888.

[63] N. Feamster, J. Rexford, and E. Zegura, "The road to SDN: An intellectual history of programmable networks," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 44, no. 2, pp. 87–98, Apr. 2014, doi: 10.1145/2602204.2602219.

[64] H. Feng, D. Chen, H. Lv, and Z. Lv, "Game theory in network security for digital twins in industry," *Digit. Commun. Netw.*, vol. 10, no. 4, pp. 1068–1078, Aug. 2024.

[65] M. Ferriol-Galmés, J. Suárez-Varela, J. Paillissé, X. Shi, S. Xiao, X. Cheng, P. Barlet-Ros, and A. Cabellos-Aparicio, "Building a digital twin for network optimization using graph neural networks," *Comput. Netw.*, vol. 217, Nov. 2022, Art. no. 109329.

[66] M. Ferriol-Galmés, K. Rusek, J. Suárez-Varela, S. Xiao, X. Shi, X. Cheng, B. Wu, P. Barlet-Ros, and A. Cabellos-Aparicio, "RouteNet-erlang: A graph neural network for network performance evaluation," in *Proc. IEEE Conf. Comput. Commun.*, May 2022, pp. 2018–2027.

[67] 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.

[68] T. Gabor, L. Belzner, M. Kiermeier, M. T. Beck, and A. Neitz, "A simulation-based architecture for smart cyber-physical systems," in *Proc. IEEE Int. Conf. Autonomic Comput. (ICAC)*, Jul. 2016, pp. 374–379.

[69] S. Ghimire, R. Melo, J. C. Ferreira, C. Agostinho, and R. Jardim-Gonçalves, "Continuous data collection framework for manufacturing industries," in *Proc. OTM Confederated Int. Conf. 'Move Meaningful Internet Syst.* Cham, Switzerland: Springer, Jan. 2015, pp. 29–40.

[70] M. Giordani, M. Polese, M. Mezzavilla, S. Rangan, and M. Zorzi, "Toward 6G networks: Use cases and technologies," *IEEE Commun. Mag.*, vol. 58, no. 3, pp. 55–61, Mar. 2020.

[71] P. K. Gkonis, P. T. Trakadas, and D. I. Kaklamani, "A comprehensive study on simulation techniques for 5G networks: State of the art results, analysis, and future challenges," *Electronics*, vol. 9, no. 3, p. 468, Mar. 2020.

[72] (2024). *GNS-3: The Software That Empowers Network Professionals*. Accessed: Sep. 11, 2024. [Online]. Available: https://www.gns3.com/

[73] A. Gohar and G. Nencioni, "The role of 5G technologies in a smart city: The case for intelligent transportation system," *Sustainability*, vol. 13, no. 9, p. 5188, May 2021.

[74] M. González, O. Salgado, J. Croes, B. Pluymers, and W. Desmet, "A digital twin for operational evaluation of vertical transportation systems," *IEEE Access*, vol. 8, pp. 114389–114400, 2020.

[75] S. Gu, D. Guo, G. Tang, L. Luo, Y. Sun, and X. Luo, "HyEdge: A cooperative edge computing framework for provisioning private and public services," *ACM Trans. Internet Things*, vol. 4, no. 2, pp. 1–23, May 2023.

[76] J. W. Guck, A. Van Bemten, M. Reisslein, and W. Kellerer, "Unicast QoS routing algorithms for SDN: A comprehensive survey and performance evaluation," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 1, pp. 388–415, 1st Quart., 2018.

[77] L. Guillen, H. Takahira, S. Izumi, T. Abe, and T. Suganuma, "On designing a resilient SDN C/M-plane for multi-controller failure in disaster situations," *IEEE Access*, vol. 8, pp. 141719–141732, 2020, doi: 10.1109/ACCESS.2020.3013323.

[78] Q. Guo, F. Tang, T. K. Rodrigues, and N. Kato, "Five disruptive technologies in 6G to support digital twin networks," *IEEE Wireless Commun.*, vol. 31, no. 1, pp. 149–155, Jan. 2023.

[79] Y. Guo, Z. Wang, X. Yin, X. Shi, and J. Wu, "Optimize routing in hybrid SDN network with changing traffic," in *Proc. 26th Int. Conf. Comput. Commun. Netw. (ICCCN)*, Jul. 2017, pp. 1–8, doi: 10.1109/ICCCN.2017.8038397.

[80] Y. Guo, Z. Wang, X. Yin, X. Shi, and J. Wu, "Traffic engineering in SDN/OSPF hybrid network," in *Proc. IEEE 22nd Int. Conf. Netw. Protocols*, Oct. 2014, pp. 563–568.

[81] A. Hakiri, A. Gokhale, S. B. Yahia, and N. Mellouli, "A comprehensive survey on digital twin for future networks and emerging Internet of Things industry," *Comput. Netw.*, vol. 244, May 2024, Art. no. 110350.

[82] G. Han, J. Jiang, N. Bao, L. Wan, and M. Guizani, "Routing protocols for underwater wireless sensor networks," *IEEE Commun. Mag.*, vol. 53, no. 11, pp. 72–78, Nov. 2015.

[83] N. Hassan, K. A. Yau, and C. Wu, "Edge computing in 5G: A review," *IEEE Access*, vol. 7, pp. 127276–127289, 2019.

[84] Y. He, M. Yang, Z. He, and M. Guizani, "Resource allocation based on digital twin-enabled federated learning framework in heterogeneous cellular network," *IEEE Trans. Veh. Technol.*, vol. 72, no. 1, pp. 1149–1158, Jan. 2023.

[85] L. Hogie, I. Tahiri, D. Papadimitriou, and F. Majorczyk, "DRMSim: A network simulator for the investigation of routing schemes user manual," Institut National de Recherche en Informatique et en Automatique (INRIA), Sophia Antipolis, France, Tech. Rep., 2010.

[86] C. Hu, W. Fan, E. Zeng, Z. Hang, F. Wang, L. Qi, and M. Z. A. Bhuiyan, "Digital twin-assisted real-time traffic data prediction method for 5G-enabled Internet of Vehicles," *IEEE Trans. Ind. Informat.*, vol. 18, no. 4, pp. 2811–2819, Apr. 2022.

[87] L. Hu, N.-T. Nguyen, W. Tao, M. C. Leu, X. F. Liu, M. R. Shahriar, and S. M. N. Al Sunny, "Modeling of cloud-based digital twins for smart manufacturing with MT connect," *Proc. Manuf.*, vol. 26, pp. 1193–1203, Jan. 2018.

[88] H.-X. Hu, Y. Feng, Q. Hu, and Y. Zhang, "A masked one-dimensional convolutional autoencoder for bearing fault diagnosis based on digital twin enabled industrial Internet of Things," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 10, pp. 3242–3253, Oct. 2023.

[89] X. Hu, S. Li, T. Huang, B. Tang, R. Huai, and L. Chen, "How simulation helps autonomous driving: A survey of sim2real, digital twins, and parallel intelligence," *IEEE Trans. Intell. Vehicles*, vol. 9, no. 1, pp. 593–612, Jan. 2024.

[90] Z. Hu, S. Lou, Y. Xing, X. Wang, D. Cao, and C. Lv, "Review and perspectives on driver digital twin and its enabling technologies for intelligent vehicles," *IEEE Trans. Intell. Vehicles*, vol. 7, no. 3, pp. 417–440, Sep. 2022.

[91] M. S. Irfan, S. Dasgupta, and M. Rahman, "Toward transportation digital twin systems for traffic safety and mobility: A review," *IEEE Internet Things J.*, vol. 11, no. 14, pp. 24581–24603, Jul. 2024.

[92] S. Ivanov, K. Nikolskaya, G. Radchenko, L. Sokolinsky, and M. Zymbler, "Digital twin of city: Concept overview," in *Proc. Global Smart Ind. Conf. (GloSIC)*, Nov. 2020, pp. 178–186.

[93] M. Javaid, A. Haleem, and R. Suman, "Digital twin applications toward industry 4.0: A review," *Cognit. Robot.*, vol. 3, pp. 71–92, Jan. 2023.

[94] X. Ji, W. Yue, C. Li, Y. Chen, N. Xue, and Z. Sha, "Digital twin empowered model free prediction of accident-induced congestion in urban road networks," in *Proc. IEEE 95th Veh. Technol. Conf. (VTC-Spring)*, Jun. 2022, pp. 1–6.

[95] P. Jia and X. Wang, "A new virtual network topology-based digital twin for spatial–temporal load-balanced user association in 6G HetNets," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 10, pp. 3080–3094, Oct. 2023.

[96] P. Jia, X. Wang, and X. Shen, "Accurate and efficient digital twin construction using concurrent end-to-end synchronization and multi-attribute data resampling," *IEEE Internet Things J.*, vol. 10, no. 6, pp. 4857–4870, Mar. 2023.

[97] S. Kaffash, A. T. Nguyen, and J. Zhu, "Big data algorithms and applications in intelligent transportation system: A review and bibliometric analysis," *Int. J. Prod. Econ.*, vol. 231, Jan. 2021, Art. no. 107868.

[98] D. Kanellopoulos and V. Sharma, "Dynamic load balancing techniques in the IoT: A review," *Symmetry*, vol. 14, no. 12, p. 2554, Dec. 2022.

[99] D. Kanellopoulos, V. K. Sharma, T. Panagiotakopoulos, and A. Kameas, "Networking architectures and protocols for IoT applications in smart cities: Recent developments and perspectives," *Electronics*, vol. 12, no. 11, p. 2490, May 2023.

[100] K. Kang, S. Belkhale, G. Kahn, P. Abbeel, and S. Levine, "Generalization through simulation: Integrating simulated and real data into deep reinforcement learning for vision-based autonomous flight," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 6008–6014.

[101] A. Karamchandani, M. Sanz, Á. Burgaleta, L. De La Cal, A. Mozo, J. I. Moreno, A. Pastor, and D. R. López, "Design of an AI-driven network digital twin for advanced 5G-6G network management," in *Proc. IEEE Netw. Operations Manage. Symp.*, May 2024, pp. 1–7.

[102] K. Kaur, J. Singh, and N. S. Ghumman, "Mininet as software defined networking testing platform," in *Proc. Int. Conf. Commun., Comput. Syst. (ICCCS))*, 2014, pp. 139–142.

[103] S. Keshav, *REAL: A Network Simulator*. Berkeley, CA, USA: University of California, 1988.

[104] L. U. Khan, W. Saad, D. Niyato, Z. Han, and C. S. Hong, "Digital-twin-enabled 6G: Vision, architectural trends, and future directions," *IEEE Commun. Mag.*, vol. 60, no. 1, pp. 74–80, Jan. 2022.

[105] L. U. Khan, W. Saad, Z. Han, E. Hossain, and C. S. Hong, "Federated learning for Internet of Things: Recent advances, taxonomy, and open challenges," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 3, pp. 1759–1799, 3rd Quart., 2021.

[106] M. Kherbache, M. Maimour, and E. Rondeau, "Network digital twin for the industrial Internet of Things," in *Proc. IEEE 23rd Int. Symp. World Wireless, Mobile Multimedia Netw. (WoWMoM)*, Jun. 2022, pp. 573–578.

[107] M. Kherbache, M. Maimour, and E. Rondeau, "When digital twin meets network softwarization in the industrial IoT: Real-time requirements case study," *Sensors*, vol. 21, no. 24, p. 8194, Dec. 2021.

[108] M. Kherbache, A. Ahmed, M. Maimour, and E. Rondeau, "Constructing a network digital twin through formal modeling: Tackling the virtual–real mapping challenge in IIoT networks," *Internet Things*, vol. 24, Dec. 2023, Art. no. 101000.

[109] S. Khorsandroo, A. G. Sánchez, A. S. Tosun, J. Arco, and R. Doriguzzi-Corin, "Hybrid SDN evolution: A comprehensive survey of the state-of-the-art," *Comput. Netw.*, vol. 192, Jun. 2021, Art. no. 107981, doi: 10.1016/j.comnet.2021.107981.

[110] J. Konečný, H. B. McMahan, D. Ramage, and P. Richtárik, "Federated optimization: Distributed machine learning for on-device intelligence," 2016, *arXiv:1610.02527*.

[111] X. Kong, Y. Wu, H. Wang, and F. Xia, "Edge computing for Internet of Everything: A survey," *IEEE Internet Things J.*, vol. 9, no. 23, pp. 23472–23485, Dec. 2022.

[112] S. A. P. Kumar, R. Madhumathi, P. R. Chelliah, L. Tao, and S. Wang, "A novel digital twin-centric approach for driver intention prediction and traffic congestion avoidance," *J. Reliable Intell. Environ.*, vol. 4, no. 4, pp. 199–209, Dec. 2018.

[113] J. Kuprat, Y. Pascal, and M. Liserre, "Real-time thermal characterization of power semiconductors using a PSO-based digital twin approach," in *Proc. 24th Eur. Conf. Power Electron. Appl. (EPE ECCE Europe)*, Sep. 2022, pp. P.1–P.8.

[114] F. Laamarti, H. F. Badawi, Y. Ding, F. Arafsha, B. Hafidh, and A. E. Saddik, "An ISO/IEEE 11073 standardized digital twin framework for health and well-being in smart cities," *IEEE Access*, vol. 8, pp. 105950–105961, 2020.

[115] J. Lai, Z. Chen, J. Zhu, W. Ma, L. Gan, S. Xie, and G. Li, "Deep learning based traffic prediction method for digital twin network," *Cognit. Comput.*, vol. 15, no. 5, pp. 1748–1766, Sep. 2023.

[116] R. Larsson, "Creating digital twin distributed networks using switches with programmable data plane," Dept. Comput. Inf. Sci., Linköping Univ., Linköping, Sweden, Tech. Rep. LIU-IDA/LITH-EX-A–2021/009–SE, 2021.

[117] B. P. L. Lau, S. H. Marakkalage, Y. Zhou, N. U. Hassan, C. Yuen, M. Zhang, and U.-X. Tan, "A survey of data fusion in smart city applications," *Inf. Fusion*, vol. 52, pp. 357–374, Dec. 2019.

[118] H. Lehner and L. Dorffner, "Digital geoTwin vienna: Towards a digital twin city as geodata hub," *PFG-J. Photogramm., Remote Sens. Geoinf. Sci.*, vol. 88, no. 1, pp. 63–75, Feb. 2020.

[119] P. Levis and N. Lee, "TOSSIM: A simulator for TinyOS networks," *UC Berkeley*, vol. 24, p. 99, Jan. 2003.

[120] B. Li, Y. Liu, L. Tan, H. Pan, and Y. Zhang, "Digital twin assisted task offloading for aerial edge computing and networks," *IEEE Trans. Veh. Technol.*, vol. 71, no. 10, pp. 10863–10877, Oct. 2022.

[121] L. Li, L. Tang, Q. Liu, Y. Wang, X. He, and Q. Chen, "DTN assisted dynamic cooperative slicing for delay-sensitive service in MEC-enabled IoT via deep deterministic policy gradient with variable action," *IEEE Internet Things J.*, vol. 10, no. 12, pp. 10708–10724, Dec. 2023.

[122] L. Li, X. Chen, Y. Xie, J. Wang, G. Lv, J. Li, X. Wang, and B. Wu, "Anomaly detection of internet service quality degradation in digital twin for fixed access network," in *Proc. 10th Int. Conf. Inf. Syst. Comput. Technol. (ISCTech)*, Dec. 2022, pp. 402–408.

[123] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *IEEE Signal Process. Mag.*, vol. 37, no. 3, pp. 50–60, May 2020.

[124] Y. Li and M. Chen, "Software-defined network function virtualization: A survey," *IEEE Access*, vol. 3, pp. 2542–2553, 2015, doi: 10.1109/ACCESS.2015.2499271.

[125] Z. Li, M. Duan, B. Xiao, and S. Yang, "A novel anomaly detection method for digital twin data using deconvolution operation with attention mechanism," *IEEE Trans. Ind. Informat.*, vol. 19, no. 5, pp. 7278–7286, May 2023.

[126] Y. Liang, Z. Yin, L. Nie, and Y. Ba, "Shared steering control with predictive risk field enabled by digital twin," *IEEE Trans. Intell. Vehicles*, vol. 8, no. 5, pp. 3256–3269, May 2023.

[127] X. Liao, J. Shi, Z. Li, L. Zhang, and B. Xia, "A model-driven deep reinforcement learning heuristic algorithm for resource allocation in ultra-dense cellular networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 1, pp. 983–997, Jan. 2020.

[128] X. Liao, Z. Wang, X. Zhao, K. Han, P. Tiwari, M. J. Barth, and G. Wu, "Cooperative ramp merging design and field implementation: A digital twin approach based on vehicle-to-cloud communication," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 5, pp. 4490–4500, May 2022.

[129] W. Y. B. Lim, N. C. Luong, D. T. Hoang, Y. Jiao, Y.-C. Liang, Q. Yang, D. Niyato, and C. Miao, "Federated learning in mobile edge networks: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 3, pp. 2031–2063, 3rd Quart., 2020.

[130] D. Liu, K. Guo, B. Wang, and Y. Peng, "Summary and perspective survey on digital twin technology," *Chin. J. Sci. Instrum.*, vol. 39, no. 11, pp. 1–10, 2018.

[131] D. Liu, Y. Du, W. Chai, C. Lu, and M. Cong, "Digital twin and data-driven quality prediction of complex die-casting manufacturing," *IEEE Trans. Ind. Informat.*, vol. 18, no. 11, pp. 8119–8128, Nov. 2022.

[132] K. Liu, L. Song, W. Han, Y. Cui, and Y. Wang, "Time-varying error prediction and compensation for movement axis of CNC machine tool based on digital twin," *IEEE Trans. Ind. Informat.*, vol. 18, no. 1, pp. 109–118, Jan. 2022.

[133] Q. Liu, L. Tang, T. Wu, and Q. Chen, "Deep reinforcement learning for resource demand prediction and virtual function network migration in digital twin network," *IEEE Internet Things J.*, vol. 10, no. 21, pp. 19102–19116, Feb. 2023.

[134] T. Liu, L. Tang, W. Wang, Q. Chen, and X. Zeng, "Digital-twin-assisted task offloading based on edge collaboration in the digital twin edge network," *IEEE Internet Things J.*, vol. 9, no. 2, pp. 1427–1444, Jan. 2022.

[135] W. Liu, M. A. Hossain, and N. Ansari, "Mobile edge computing for multi-services digital twin-enabled IoT heterogeneous networks," *IEEE Trans. Cognit. Commun. Netw.*, early access, Nov. 4, 2024, doi: 10.1109/TCCN.2024.3490779.

[136] W. Liu, B. Li, W. Xie, Y. Dai, and Z. Fei, "Energy efficient computation offloading in aerial edge networks with multi-agent cooperation," *IEEE Trans. Wireless Commun.*, vol. 22, no. 9, pp. 5725–5739, Sep. 2023.

[137] Y. Liu, A. Yang, Q. Zeng, Y. Sun, J. Gao, and Z. Lv, "Task scheduling of real-time traffic information processing based on digital twins," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 11, pp. 13171–13179, Nov. 2022.

[138] Y. Liu, Z. Wang, K. Han, Z. Shou, P. Tiwari, and J. H. L. Hansen, "Sensor fusion of camera and cloud digital twin information for intelligent vehicles," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Oct. 2020, pp. 182–187.

[139] Y. Lu, X. Huang, K. Zhang, S. Maharjan, and Y. Zhang, "Communication-efficient federated learning for digital twin edge networks in industrial IoT," *IEEE Trans. Ind. Informat.*, vol. 17, no. 8, pp. 5709–5718, Aug. 2021, doi: 10.1109/TII.2020.3010798.

[140] Y. Lu, X. Qiu, and Y. Xing, "Digital twin-based operation simulation system and application framework for electromechanical products," in *Proc. Int. Conf. Comput., Control Robot. (ICCCR)*, Jan. 2021, pp. 146–150.

[141] Y. Lu, S. Maharjan, and Y. Zhang, "Adaptive edge association for wireless digital twin networks in 6G," *IEEE Internet Things J.*, vol. 8, no. 22, pp. 16219–16230, Nov. 2021.

[142] Y. Lu, X. Huang, K. Zhang, S. Maharjan, and Y. Zhang, "Communication-efficient federated learning and permissioned blockchain for digital twin edge networks," *IEEE Internet Things J.*, vol. 8, no. 4, pp. 2276–2288, Feb. 2021.

[143] Y. Lu, X. Huang, K. Zhang, S. Maharjan, and Y. Zhang, "Low-latency federated learning and blockchain for edge association in digital twin empowered 6G networks," *IEEE Trans. Ind. Informat.*, vol. 17, no. 7, pp. 5098–5107, Jul. 2021.

[144] Z. Lu and H. Yang, *Unlocking the Power of OPNET Modeler*. Cambridge, MA, USA: Cambridge Univ. Press, 2012.

[145] W. Luo, T. Hu, C. Zhang, and Y. Wei, "Digital twin for CNC machine tool: Modeling and using strategy," *J. Ambient Intell. Humanized Comput.*, vol. 10, no. 3, pp. 1129–1140, Mar. 2019.

[146] Z. Lv, Y. Li, H. Feng, and H. Lv, "Deep learning for security in digital twins of cooperative intelligent transportation systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 9, pp. 16666–16675, Sep. 2022.

[147] M. Maimour, A. Ahmed, and E. Rondeau, "Survey on digital twins for natural environments: A communication network perspective," *Internet Things*, vol. 25, Apr. 2024, Art. no. 101070.

[148] B. Mao, Y. Kawamoto, and N. Kato, "AI-based joint optimization of QoS and security for 6G energy harvesting Internet of Things," *IEEE Internet Things J.*, vol. 7, no. 8, pp. 7032–7042, Aug. 2020.

[149] B. Mao, Z. Md. Fadlullah, F. Tang, N. Kato, O. Akashi, T. Inoue, and K. Mizutani, "Routing or computing? The paradigm shift towards intelligent computer network packet transmission based on deep learning," *IEEE Trans. Comput.*, vol. 66, no. 11, pp. 1946–1960, Nov. 2017.

[150] J. M. Martínez Caro, J. Pérez Romero, R. Agüero Calvo, A. Muñiz, E. Atxutegi, L. Díez, L. M. Contreras, I. González, F. J. Moreno-Muro, and P. Pavón-Marino, "The role of network digital twinning in a 6G-oriented optimization-as-a-service platform," in *Libro de actas del 38th Simposio Nacional de la Unión Científica Internacional de Radio*, 2023, pp. 1–4.

[151] A. Masaracchia, V. Sharma, B. Canberk, O. A. Dobre, and T. Q. Duong, "Digital twin for 6G: Taxonomy, research challenges, and the road ahead," *IEEE Open J. Commun. Soc.*, vol. 3, pp. 2137–2150, 2022.

[152] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Y. Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. Artif. Intell. Statist.*, Jan. 2016, pp. 1273–1282.

[153] J. Koneč ný, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," in *Proc. 29th Conf. Neural Inf. Process. Syst. (NIPS)*, Barcelona, Spain, Jan. 2016, pp. 5–10.

[154] A. Mendiola, J. Astorga, E. Jacob, and M. Higuero, "A survey on the contributions of software-defined networking to traffic engineering," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 2, pp. 918–953, 2nd Quart., 2017, doi: 10.1109/COMST.2016.2633579.

[155] A. Mestres, E. Alarcón, Y. Ji, and A. Cabellos-Aparicio, "Understanding the modeling of computer network delays using neural networks," in *Proc. Workshop Big Data Anal. Mach. Learn. Data Commun. Netw.*, Aug. 2018, pp. 46–52.

[156] M. Mezzavilla, M. Zhang, M. Polese, R. Ford, S. Dutta, S. Rangan, and M. Zorzi, "End-to-end simulation of 5G mmWave networks," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 3, pp. 2237–2263, 3rd Quart., 2018.

[157] S. Mihai, M. Yaqoob, D. V. Hung, W. Davis, P. Towakel, M. Raza, M. Karamanoglu, B. Barn, D. Shetve, R. V. Prasad, H. Venkataraman, R. Trestian, and H. X. Nguyen, "Digital twins: A survey on enabling technologies, challenges, trends and future prospects," *IEEE Commun. Surveys Tuts.*, vol. 24, no. 4, pp. 2255–2291, 4th Quart., 2022.

[158] M. Milton, H. L. Ginn, and A. Benigni, "Controller-embeddable probabilistic real-time digital twins for power electronic converter diagnostics," *IEEE Trans. Power Electron.*, vol. 35, no. 9, pp. 9850–9864, Sep. 2020.

[159] R. Minerva, G. M. Lee, and N. Crespi, "Digital twin in the IoT context: A survey on technical features, scenarios, and architectural models," *Proc. IEEE*, vol. 108, no. 10, pp. 1785–1824, Oct. 2020.

[160] N. Mohammadi and J. E. Taylor, "Smart city digital twins," in *Proc. IEEE Symp. Ser. Comput. Intell. (SSCI)*, Nov. 2017, pp. 1–5.

[161] S. Si-Mohammed, A. Bardou, T. Begin, I. Guérin Lassous, and P. Vicat-Blanc, "NS+NDT: Smart integration of network simulation in network digital twin, application to IoT networks," *Future Gener. Comput. Syst.*, vol. 157, pp. 124–144, Aug. 2024.

[162] J. T. Moy, *OSPF: Anatomy of an Internet Routing Protocol*. Reading, MA, USA: Addison-Wesley, 1998.

[163] T. Mukherjee and T. DebRoy, "A digital twin for rapid qualification of 3D printed metallic components," *Appl. Mater. Today*, vol. 14, pp. 59–65, Mar. 2019.

[164] G. Mylonas, A. Kalogeras, G. Kalogeras, C. Anagnostopoulos, C. Alexakos, and L. Muñoz, "Digital twins from smart manufacturing to smart cities: A survey," *IEEE Access*, vol. 9, pp. 143222–143249, 2021.

[165] H. Nan, R. Li, X. Zhu, J. Ma, and D. Niyato, "An efficient data-driven traffic prediction framework for network digital twin," *IEEE Netw.*, vol. 38, no. 1, pp. 22–29, Jan. 2024, doi: 10.1109/MNET.2023.3335952.

[166] A. Nanda, D. Puthal, J. J. P. C. Rodrigues, and S. A. Kozlov, "Internet of Autonomous Vehicles communications security: Overview, issues, and directions," *IEEE Wireless Commun.*, vol. 26, no. 4, pp. 60–65, Aug. 2019.

[167] A. Nasrallah, A. S. Thyagaturu, Z. Alharbi, C. Wang, X. Shao, M. Reisslein, and H. ElBakoury, "Ultra-low latency (ULL) networks: The IEEE TSN and IETF DetNet standards and related 5G ULL research," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 1, pp. 88–145, 1st Quart., 2019.

[168] H. X. Nguyen, R. Trestian, D. To, and M. Tatipamula, "Digital twin for 5G and beyond," *IEEE Commun. Mag.*, vol. 59, no. 2, pp. 10–15, Feb. 2021.

[169] L. Nie, X. Wang, Q. Zhao, Z. Shang, L. Feng, and G. Li, "Digital twin for transportation big data: A reinforcement learning-based network traffic prediction approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 1, pp. 896–906, Jan. 2023.

[170] T. Nochta, L. Wan, J. M. Schooling, and A. K. Parlikad, "A socio-technical perspective on urban analytics: The case of city-scale digital twins," *J. Urban Technol.*, vol. 28, nos. 1–2, pp. 263–287, Apr. 2021.

[171] W. B. Nordgren, "Flexsim simulation environment," in *Proc. Winter Simulation Conf.*, vol. 1, 2003, pp. 250–252.

[172] M. O. Ozdogan, L. Carkacioglu, and B. Canberk, "Digital twin driven blockchain based reliable and efficient 6G edge network," in *Proc. 18th Int. Conf. Distrib. Comput. Sensor Syst. (DCOSS)*, May 2022, pp. 342–348.

[173] J. Pan and R. Jain, "A survey of network simulation tools: Current status and future developments," Washington Univ. St. Louis, St. Louis, MO, USA, Tech. Rep. WUSTL/CSE 567, 2008. [Online]. Available: http://www.cs.wustl.edu/~jain/cse567-08/ftp/simtools/index.html

[174] Y. H. Pan, N. Q. Wu, T. Qu, P. Z. Li, K. Zhang, and H. F. Guo, "Digital-twin-driven production logistics synchronization system for vehicle routing problems with pick-up and delivery in industrial park," *Int. J. Comput. Integr. Manuf.*, vol. 34, nos. 7–8, pp. 814–828, Aug. 2021.

[175] B. K. Pandey, M. A. Paramashivan, D. Pandey, A. S. George, A. Agarwal, D. A. Mahajan, P. D. Dadheech, and S. Pramanik, "Future directions of digital twin architectures for 6G communication networks," in *Security Issues and Solutions in 6G Communications and Beyond*. Hershey, PA, USA: IGI Global, 2024, pp. 197–211.

[176] J. Park, S. Samarakoon, M. Bennis, and M. Debbah, "Wireless network intelligence at the edge," *Proc. IEEE*, vol. 107, no. 11, pp. 2204–2239, Nov. 2019.

[177] R. L. Patel, M. J. Pathak, and A. J. Nayak, "Survey on network simulators," *Int. J. Comput. Appl.*, vol. 182, no. 21, pp. 23–30, 2018.

[178] *Philips Digital Twin of Care*. Accessed: Aug. 12, 2024. [Online]. Available: https://www.philips.co.in/healthcare/consulting/technology-transformation-analytics/healthcare-modeling-simulation-consulting

[179] M. Piórkowski, M. Raya, A. L. Lugo, P. Papadimitratos, M. Grossglauser, and J.-P. Hubaux, "TraNS: Realistic joint traffic and network simulator for VANETs," *ACM SIGMOBILE Mobile Comput. Commun. Rev.*, vol. 12, no. 1, pp. 31–33, Jan. 2008.

[180] F. Pires, B. Ahmad, A. P. Moreira, and P. Leitão, "Digital twin based what-if simulation for energy management," in *Proc. 4th IEEE Int. Conf. Ind. Cyber-Phys. Syst. (ICPS)*, May 2021, pp. 309–314.

[181] R. Poorzare and A. C. Augé, "Challenges on the way of implementing TCP over 5G networks," *IEEE Access*, vol. 8, pp. 176393–176415, 2020, doi: 10.1109/ACCESS.2020.3026540.

[182] R. Poorzare and A. C. Augé, "How sufficient is TCP when deployed in 5G mmWave networks over the urban deployment?" *IEEE Access*, vol. 9, pp. 36342–36355, 2021.

[183] J. Pushpa and S. Kalyani, "Using fog computing/edge computing to leverage digital twin," in *Advances in Computers*, vol. 117. Amsterdam, The Netherlands: Elsevier, 2020, pp. 51–77.

[184] Q. Qi, F. Tao, T. Hu, N. Anwer, A. Liu, Y. Wei, L. Wang, and A. Y. C. Nee, "Enabling technologies and tools for digital twin," *J. Manuf. Syst.*, vol. 58, pp. 3–21, Jan. 2021.

[185] Q. Qi, D. Zhao, T. W. Liao, and F. Tao, "Modeling of cyber-physical systems and digital twin based on edge computing, fog computing and cloud computing towards smart manufacturing," in *Proc. Int. Manuf. Sci. Eng. Conf.*, vol. 51357, 2018, p. 1.

[186] C. Qian, X. Liu, C. Ripley, M. Qian, F. Liang, and W. Yu, "Digital twin—Cyber replica of physical things: Architecture, applications and future research directions," *Future Internet*, vol. 14, no. 2, p. 64, Feb. 2022.

[187] B. Qin, H. Pan, Y. Dai, X. Si, X. Huang, C. Yuen, and Y. Zhang, "Machine and deep learning for digital twin networks: A survey," *IEEE Internet Things J.*, vol. 11, no. 21, pp. 34694–34716, Nov. 2024.

[188] L. Raes, P. Michiels, T. Adolphi, C. Tampere, A. Dalianis, S. McAleer, and P. Kogut, "DUET: A framework for building interoperable and trusted digital twins of smart cities," *IEEE Internet Comput.*, vol. 26, no. 3, pp. 43–50, May 2022.

[189] D. R. R. Raj, T. A. Shaik, A. Hirwe, P. Tammana, and K. Kataoka, "Building a digital twin network of SDN using knowledge graphs," *IEEE Access*, vol. 11, pp. 63092–63106, 2023, doi: 10.1109/ACCESS.2023.3288813.

[190] A. Rasheed, O. San, and T. Kvamsdal, "Digital twin: Values, challenges and enablers from a modeling perspective," *IEEE Access*, vol. 8, pp. 21980–22012, 2020.

[191] A. Rassõlkin, T. Vaimann, A. Kallaste, and V. Kuts, "Digital twin for propulsion drive of autonomous electric vehicle," in *Proc. IEEE 60th Int. Sci. Conf. Power Electr. Eng. Riga Tech. Univ. (RTUCON)*, Oct. 2019, pp. 1–4.

[192] A. K. Rathi and A. J. Santiago, "The new NETSIM simulation program," *Traffic Eng. Control*, vol. 31, no. 5, pp. 317–320, May 1990.

[193] M. M. Rathore, S. A. Shah, D. Shukla, E. Bentafat, and S. Bakiras, "The role of AI, machine learning, and big data in digital twinning: A systematic literature review, challenges, and opportunities," *IEEE Access*, vol. 9, pp. 32030–32052, 2021.

[194] (2022). *Recommendation ITU-T Y.3090: Digital Twin Network—Requirements and Architecture*. Accessed: Jul. 17, 2024. [Online]. Available: https://handle.itu.int/11.1002/1000/14852

[195] J. Reis, M. Rocha, T. K. Phan, D. Griffin, F. Le, and M. Rio, "Deep neural networks for network routing," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2019, pp. 1–8, doi: 10.1109/IJCNN.2019.8851733.

[196] A. Ricci, A. Croatti, and S. Montagna, "Pervasive and connected digital twins—A vision for digital health," *IEEE Internet Comput.*, vol. 26, no. 5, pp. 26–32, Sep. 2022.

[197] T. M. Ricks, T. E. Lacy, E. J. Pineda, B. A. Bednarcyk, and S. M. Arnold, "Computationally efficient solution of the high-fidelity generalized method of cells micromechanics relations," in *Proc. Amer. Soc. Compos. 30th Annu. Tech. Conf.*, East Lansing, MI, USA, Nov. 2015, pp. 28–30.

[198] G. F. Riley and T. R. Henderson, "The NS-3 network simulator," in *Modeling and Tools for Network Simulation*. Berlin, Germany: Springer, 2010, pp. 15–34, doi: 10.1007/978-3-642-12331-3_2.

[199] M. Sanz Rodrigo, D. Rivera, J. I. Moreno, M. Àlvarez-Campana, and D. R. López, "Digital twins for 5G networks: A modeling and deployment methodology," *IEEE Access*, vol. 11, pp. 38112–38126, 2023.

[200] M. W. Rohrer and I. W. McGregor, "Simulating reality using AutoMod," in *Proc. Winter Simulation Conf.*, vol. 1, 2002, pp. 173–181.

[201] J. Ros-Giralt, N. Amsel, S. Yellamraju, J. Ezick, R. Lethin, Y. Jiang, A. Feng, L. Tassiulas, Z. Wu, M. Y. Teh, and K. Bergman, "Designing data center networks using bottleneck structures," in *Proc. ACM SIGCOMM Conf.*, Aug. 2021, pp. 319–348.

[202] J. Ros-Giralt, A. Bohara, S. Yellamraju, H. Langston, Y. Jiang, L. Tassiulas, J. Li, Y. Tan, and M. Veeraraghavan, "On the bottleneck structure of congestion-controlled networks," *ACM SIGMETRICS Perform. Eval. Rev.*, vol. 48, no. 1, pp. 67–68, Jul. 2020.

[203] T. Ruohomäki, E. Airaksinen, P. Huuska, O. Kesäniemi, M. Martikka, and J. Suomisto, "Smart city platform enabling digital twin," in *Proc. Int. Conf. Intell. Syst. (IS)*, Sep. 2018, pp. 155–161.

[204] K. Rusek, J. Suárez-Varela, P. Almasan, P. Barlet-Ros, and A. Cabellos-Aparicio, "RouteNet: Leveraging graph neural networks for network modeling and optimization in SDN," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 10, pp. 2260–2270, Oct. 2020.

[205] Y. Sahni, J. Cao, S. Zhang, and L. Yang, "Edge mesh: A new paradigm to enable distributed intelligence in Internet of Things," *IEEE Access*, vol. 5, pp. 16441–16458, 2017.

[206] A. M. V. V. Sai, C. Wang, Z. Cai, and Y. Li, "Navigating the digital twin network landscape: A survey on architecture, applications, privacy and security," *High-Confidence Comput.*, vol. 4, no. 4, Dec. 2024, Art. no. 100269.

[207] S. B. Hadj Said, M.-T. Thi, M. Kellil, and A. Olivereau, "On the management of TSN networks in 6G: A network digital twin approach," in *Proc. IEEE 28th Int. Conf. Emerg. Technol. Factory Autom. (ETFA)*, Sep. 2023, pp. 1–7.

[208] K. Salah, M. H. U. Rehman, N. Nizamuddin, and A. Al-Fuqaha, "Blockchain for AI: Review and open research challenges," *IEEE Access*, vol. 7, pp. 10127–10149, 2019.

[209] J. J. Lee, "Simulation of hybrid computer architectures: Simulators, methodologies and recommendations," in *Proc. IFIP Int. Conf. Very Large Scale Integr.*, Oct. 2007, pp. 157–162.

[210] M. Schluse, M. Priggemeyer, L. Atorf, and J. Rossmann, "Experimentable digital twins—Streamlining simulation-based systems engineering for industry 4.0," *IEEE Trans. Ind. Informat.*, vol. 14, no. 4, pp. 1722–1731, Apr. 2018.

[211] G. Schroeder, C. Steinmetz, C. E. Pereira, I. Müller, N. Garcia, D. Espindola, and R. Rodrigues, "Visualising the digital twin using web services and augmented reality," in *Proc. IEEE 14th Int. Conf. Ind. Informat. (INDIN)*, Jul. 2016, pp. 522–527.

[212] G. N. Schroeder, C. Steinmetz, C. E. Pereira, and D. B. Espindola, "Digital twin data modeling with AutomationML and a communication methodology for data exchange," *IFAC-PapersOnLine*, vol. 49, no. 30, pp. 12–17, 2016.

[213] G. Schrotter and C. Hürzeler, "The digital twin of the city of Zurich for urban planning," *PFG-J. Photogramm., Remote Sens. Geoinf. Sci.*, vol. 88, no. 1, pp. 99–112, Feb. 2020.

[214] R. Sell, M. Leier, A. Rassõlkin, and J.-P. Ernits, "Autonomous last mile shuttle ISEAUTO for education and research," *Int. J. Artif. Intell. Mach. Learn.*, vol. 10, no. 1, pp. 18–30, Jan. 2020.

[215] A. Sharma, E. Kosasih, J. Zhang, A. Brintrup, and A. Calinescu, "Digital twins: State of the art theory and practice, challenges, and open research questions," *J. Ind. Inf. Integr.*, vol. 30, Nov. 2022, Art. no. 100383.

[216] V. K. Sharma, S. S. P. Shukla, and V. Singh, "A tailored Q-learning for routing in wireless sensor networks," in *Proc.2nd IEEE Int. Conf. Parallel, Distrib. Grid Comput.*, Dec. 2012, pp. 663–668.

[217] V. K. Sharma and M. Kumar, "Adaptive energy efficient load distribution using fuzzy approach," *Adhoc Sensor Wireless Netw.*, vol. 39, pp. 123–166, Jan. 2017.

[218] M. Sheraz, T. C. Chuah, Y. L. Lee, M. M. Alam, A. Al-Habashna, and Z. Han, "A comprehensive survey on revolutionizing connectivity through artificial intelligence-enabled digital twin network in 6G," *IEEE Access*, vol. 12, pp. 49184–49215, 2024.

[219] H. Shin, S. Oh, A. Isah, I. Aliyu, J. Park, and J. Kim, "Network traffic prediction model in a data-driven digital twin network architecture," *Electronics*, vol. 12, no. 18, p. 3957, Sep. 2023.

[220] *Siemens Digital Industries Software*. Accessed: Aug. 12, 2024. [Online]. Available: https://www.sw.siemens.com/en-U.S./technology/digital-twin/

[221] S. Siraj, A. Gupta, and R. Badgujar, "Network simulation tools survey," *Int. J. Adv. Res. Comput. Commun. Eng.*, vol. 1, no. 4, pp. 199–206, 2012.

[222] V. Smith, C.-K. Chiang, M. Sanjabi, and A. Talwalkar, "Federated multi-task learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, Dec. 2017, pp. 4427–4437.

[223] A. Sobeih, J. C. Hou, L.-C. Kung, N. Li, H. Zhang, W.-P. Chen, H.-Y. Tyan, and H. Lim, "J-sim: A simulation and emulation environment for wireless sensor networks," *IEEE Wireless Commun.*, vol. 13, no. 4, pp. 104–119, Aug. 2006.

[224] R. J. Somers, J. A. Douthwaite, D. J. Wagg, N. Walkinshaw, and R. M. Hierons, "Digital-twin-based testing for cyber–physical systems: A systematic literature review," *Inf. Softw. Technol.*, vol. 156, Apr. 2023, Art. no. 107145.

[225] J. Song, Y. Kang, Q. Song, L. Guo, and A. Jamalipour, "Distributed resource optimization with blockchain security for immersive digital twin in IIoT," *IEEE Trans. Ind. Informat.*, vol. 19, no. 5, pp. 7258–7267, May 2023.

[226] X. Su, Y. Ren, Z. Cai, Y. Liang, and L. Guo, "A Q-learning based routing approach for energy efficient information transmission in wireless sensor network," *IEEE Trans. Netw. Service Manage.*, vol. 20, no. 2, pp. 1949–1961, Feb. 2022.

[227] W. Sun, W. Ma, Y. Zhou, and Y. Zhang, "An introduction to digital twin standards," *GetMobile, Mobile Comput. Commun.*, vol. 26, no. 3, pp. 16–22, Oct. 2022, doi: 10.1145/3568113.3568119.

[228] W. Sun, S. Lei, L. Wang, Z. Liu, and Y. Zhang, "Adaptive federated learning and digital twin for industrial Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 17, no. 8, pp. 5605–5614, Aug. 2021.

[229] W. Sun, H. Zhang, R. Wang, and Y. Zhang, "Reducing offloading latency for digital twin edge networks in 6G," *IEEE Trans. Veh. Technol.*, vol. 69, no. 10, pp. 12240–12251, Oct. 2020.

[230] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed., Cambridge, MA, USA: MIT Press, 2018. [Online]. Available: http://incompleteideas.net/book/the-book-2nd.html

[231] F. Tang, X. Chen, T. K. Rodrigues, M. Zhao, and N. Kato, "Survey on digital twin edge networks (DITEN) toward 6G," *IEEE Open J. Commun. Soc.*, vol. 3, pp. 1360–1381, 2022.

[232] L. Tang, Y. Du, Q. Liu, J. Li, S. Li, and Q. Chen, "Digital twin assisted resource allocation for network slicing in industry 4.0 and beyond using distributed deep reinforcement learning," *IEEE Internet Things J.*, vol. 10, no. 19, pp. 16989–17006, Oct. 2023.

[233] F. Tao, Y. Zhang, Y. Cheng, J. Ren, D. Wang, Q. Qi, and P. Li, "Digital twin and blockchain enhanced smart manufacturing service collaboration and management," *J. Manuf. Syst.*, vol. 62, pp. 903–914, Jan. 2022.

[234] F. Tao, H. Zhang, A. Liu, and A. Y. C. Nee, "Digital twin in industry: State-of-the-art," *IEEE Trans. Ind. Informat.*, vol. 15, no. 4, pp. 2405–2415, Apr. 2019.

[235] F. Tao, W. Liu, M. Zhang, T.-L. Hu, Q. Qi, H. Zhang, F. Sui, T. Wang, H. Xu, and Z. Huang, "Five-dimension digital twin model and its ten applications," *Comput. Integr. Manuf. Syst.*, vol. 25, no. 1, pp. 1–18, 2019.

[236] Y. Tao, J. Wu, X. Lin, and W. Yang, "DRL-driven digital twin function virtualization for adaptive service response in 6G networks," *IEEE Netw. Lett.*, vol. 5, no. 2, pp. 125–129, Jun. 2023.

[237] Z. Tao, W. Xu, Y. Huang, X. Wang, and X. You, "Wireless network digital twin for 6G: Generative AI as a key enabler," *IEEE Wireless Commun.*, vol. 31, no. 4, pp. 24–31, Aug. 2024.

[238] M. Tariq, F. Naeem, and H. Vincent Poor, "Toward experience-driven traffic management and orchestration in digital-twin-enabled 6G networks," 2022, *arXiv:2201.04259*.

[239] A. Thapa, D. Karki, R. Guragain, S. P. Upadhyaya, B. R. Dawadi, and S. R. Joshi, "Experimental evaluation of a multi-domain routing in SDN and its inter-operability with legacy networks," in *Proc. IEEE Int. Conf. Electron., Comput. Commun. Technol. (CONECCT)*, Jul. 2021, pp. 1–6, doi: 10.1109/CONECCT52877.2021.9622615.

[240] *The Digital Power Plant: Bridging the Intelligence Gap Between People and Machines*. Accessed: Aug. 12, 2024. [Online]. Available: https://www.ge.com/digital/sites/default/files/download_assets/GE-Digital-Power-Plant-Brochure.pdf

[241] X. Tong, Q. Liu, S. Pi, and Y. Xiao, "Real-time machining data application and service based on IMT digital twin," *J. Intell. Manuf.*, vol. 31, no. 5, pp. 1113–1132, Jun. 2020.

[242] G. Top. *Strategic Technology Trends for 2020*. Accessed: Sep. 2024. [Online]. Available: https://www.gartner.com/smarterwithgartner/gartner-top-10-strategic-technology-trends-for-2020

[243] J. W. Turner, "A survey of network simulation tools and techniques," *ACM Comput. Surv.*, vol. 52, no. 4, pp. 1–36, 2020.

[244] M. Vaezi, K. Noroozi, T. D. Todd, D. Zhao, G. Karakostas, H. Wu, and X. Shen, "Digital twins from a networking perspective," *IEEE Internet Things J.*, vol. 9, no. 23, pp. 23525–23544, Dec. 2022.

[245] S. Vakaruk, A. Mozo, A. Pastor, and D. R. López, "A digital twin network for security training in 5G industrial environments," in *Proc. IEEE 1st Int. Conf. Digit. Twins Parallel Intell. (DTPI)*, Jul. 2021, pp. 395–398.

[246] A. Valadarsky, M. Schapira, D. Shahaf, and A. Tamar, "Learning to route," in *Proc. 16th ACM Workshop Hot Topics Netw.*, Nov. 2017, pp. 185–191.

[247] A. Varga and R. Hornig, "An overview of the OMNeT++ simulation environment," in *Proc. 1st Int. ICST Conf. Simulation Tools Techn. Commun. Netw. Syst.*, 2008, pp. 60–67.

[248] M. Varshney, R. Bagrodia, A. Dixit, B. Vasu, P. Aghera, and M. Srivastava, "SQualNet version 1.0 beta: Tutorial and programmers manual," Univ. California, Los Angeles (UCLA), Los Angeles, CA, USA, Tech. Rep., 2004.

[249] T. L. Veith, J. E. Kobza, and C. P. Koelling, "Netsim: Java^{TM}-based simulation for the world wide web," *Comput. Oper. Res.*, vol. 26, no. 6, pp. 607–621, May 1999.

[250] R. Verdecchia, L. Scommegna, B. Picano, M. Becattini, and E. Vicario, "Network digital twins: A systematic review," *IEEE Access*, vol. 12, pp. 145400–145416, 2024.

[251] I. Vilà, O. Sallent, and J. Pérez-Romero, "On the design of a network digital twin for the radio access network in 5G and beyond," *Sensors*, vol. 23, no. 3, p. 1197, Jan. 2023.

[252] R. Vilalta, L. Gifre, R. Casellas, R. Muñoz, R. Martínez, A. Mozo, A. Pastor, D. López, and J. P. Fernández-Palacios, "Applying digital twins to optical networks with cloud-native SDN controllers," *IEEE Commun. Mag.*, vol. 61, no. 12, pp. 128–134, Dec. 2023, doi: 10.1109/MCOM.003.2300105.

[253] (2020). *W3C. Web of Things (wot) Architecture, W3c Recommendation*. Accessed: Jul. 10, 2024. [Online]. Available: https://www.w3.org/TR/2020/REC-wot-architecture-20200409

[254] B. Wang, C. Zhang, M. Zhang, C. Liu, Z. Xie, and H. Zhang, "Digital twin analysis for driving risks based on virtual physical simulation technology," *IEEE J. Radio Freq. Identificat.*, vol. 6, pp. 938–942, 2022.

[255] D. Wang, B. Li, B. Song, Y. Liu, K. Muhammad, and X. Zhou, "Dual-driven resource management for sustainable computing in the blockchain-supported digital twin IoT," *IEEE Internet Things J.*, vol. 10, no. 8, pp. 6549–6560, Apr. 2023.

[256] D. Wang, Y. Song, Y. Zhang, X. Jiang, J. Dong, F. N. Khan, T. Sasai, S. Huang, A. P. T. Lau, M. Tornatore, and M. Zhang, "Digital twin of optical networks: A review of recent advances and future trends," *J. Lightw. Technol.*, vol. 42, no. 12, pp. 4233–4259, Jun. 15, 2024.

[257] H. Wang, Y. Wu, G. Min, and W. Miao, "A graph neural network-based digital twin for network slicing management," *IEEE Trans. Ind. Informat.*, vol. 18, no. 2, pp. 1367–1376, Feb. 2022.

[258] J. Wang, L. Ye, R. X. Gao, C. Li, and L. Zhang, "Digital twin for rotating machinery fault diagnosis in smart manufacturing," *Int. J. Prod. Res.*, vol. 57, no. 12, pp. 3920–3934, Jun. 2019.

[259] K. Wang, Z. Li, K. Nonomura, T. Yu, K. Sakaguchi, O. Hashash, and W. Saad, "Smart mobility digital twin based automated vehicle navigation system: A proof of concept," *IEEE Trans. Intell. Vehicles*, vol. 9, no. 3, pp. 4348–4361, Mar. 2024.

[260] M. Wang, Y. Lin, Q. Tian, and G. Si, "Transfer learning promotes 6G wireless communications: Recent advances and future challenges," *IEEE Trans. Rel.*, vol. 70, no. 2, pp. 790–807, Jun. 2021.

[261] M. Wang, Y. Cui, X. Wang, S. Xiao, and J. Jiang, "Machine learning for networking: Workflow, advances and opportunities," *IEEE Netw.*, vol. 32, no. 2, pp. 92–99, Mar. 2018.

[262] W. Wang, L. Tang, C. Wang, and Q. Chen, "Real-time analysis of multiple root causes for anomalies assisted by digital twin in NFV environment," *IEEE Trans. Netw. Service Manage.*, vol. 19, no. 2, pp. 905–921, Jun. 2022.

[263] W. Wang, W. He, and J. Su, "Enhancing the effectiveness of traffic engineering in hybrid SDN," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2017, pp. 1–6.

[264] Y. Wang, J. Fang, Y. Cheng, H. She, Y. Guo, and G. Zheng, "Cooperative end-edge-cloud computing and resource allocation for digital twin enabled 6G industrial IoT," *IEEE J. Sel. Topics Signal Process.*, vol. 18, no. 1, pp. 124–137, Jan. 2024.

[265] Z. Wang, K. Han, and P. Tiwari, "Digital twin-assisted cooperative driving at non-signalized intersections," *IEEE Trans. Intell. Vehicles*, vol. 7, no. 2, pp. 198–209, Jun. 2022.

[266] Z. Wang, R. Gupta, K. Han, H. Wang, A. Ganlath, N. Ammar, and P. Tiwari, "Mobility digital twin: Concept, architecture, case study, and future challenges," *IEEE Internet Things J.*, vol. 9, no. 18, pp. 17452–17467, Sep. 2022.

[267] Y. Wei, X. Zhang, L. Xie, and S. Leng, "Energy-aware traffic engineering in hybrid SDN/IP backbone networks," *J. Commun. Netw.*, vol. 18, no. 4, pp. 559–566, Aug. 2016.

[268] J. Wen, B. Gabrys, and K. Musial, "Toward digital twin oriented modeling of complex networked systems and their dynamics: A comprehensive survey," *IEEE Access*, vol. 10, pp. 66886–66923, 2022.

[269] M. Wiering and M. van Otterlo, *Reinforcement Learning: State-of-the-Art* (Adaptation, Learning, and Optimization). Berlin, Germany: Springer, 2012. [Online]. Available: https://books.google.co.in/books?id=T4wovQEACAAJ

[270] C. Wohlin, "Guidelines for snowballing in systematic literature studies and a replication in software engineering," in *Proc. 18th Int. Conf. Eval. Assessment Softw. Eng.*, May 2014, pp. 1–10.
[271] D. Wu and L. Cui, "A comprehensive survey on segment routing traffic engineering," *Digit. Commun. Netw.*, vol. 9, no. 4, pp. 990–1008, Aug. 2023, doi: 10.1016/j.dcan.2022.02.006.
[272] X. Wu, W. Lian, M. Zhou, H. Song, and H. Dong, "A digital twin-based fault diagnosis framework for bogies of high-speed trains," *IEEE J. Radio Freq. Identificat.*, vol. 7, pp. 203–207, 2023.
[273] Y. Wu, K. Zhang, and Y. Zhang, "IEEE Internet of Things journal," *GetMobile Mobile Comput. Commun.*, vol. 8, no. 18, pp. 13789–13804, Sep. 2022.
[274] Y. Wu, K. Zhang, and Y. Zhang, "Digital twin networks: A survey," *IEEE Internet Things J.*, vol. 8, no. 18, pp. 13789–13804, Sep. 2021, doi: 10.1109/JIOT.2021.3079510.
[275] C. Xenakis, A. Angelogianni, E. Veroni, E. Karapistoli, M. Ghering, N. Gerosavva, V. Machamint, P. Polvanesi, A. Brignone, J. N. Mendoza, and A. Pastor, "The SPIDER concept: A cyber range as a service platform," in *Proc. Eur. Conf. Netw. Commun. (Eucnc)*, Sep. 2020, pp. 16–17.
[276] D. Xia, J. Shi, K. Wan, J. Wan, M. Martínez-García, and X. Guan, "Digital twin and artificial intelligence for intelligent planning and energy-efficient deployment of 6G networks in smart factories," *IEEE Wireless Commun.*, vol. 30, no. 3, pp. 171–179, Jun. 2023.
[277] K. Xia, C. Sacco, M. Kirkpatrick, C. Saidy, L. Nguyen, A. Kircaliali, and R. Harik, "A digital twin to train deep reinforcement learning agent for smart manufacturing plants: Environment, interfaces and intelligence," *J. Manuf. Syst.*, vol. 58, pp. 210–230, Jan. 2021.
[278] H. Xu, J. Wu, Q. Pan, X. Guan, and M. Guizani, "A survey on digital twin for industrial Internet of Things: Applications, technologies and tools," *IEEE Commun. Surveys Tuts.*, vol. 25, no. 4, pp. 2569–2598, 4th Quart., 2023.
[279] H. Xu, J. Wu, H. Pan, J. Gu, and X. Guan, "Delay safety-aware digital twin empowered industrial sensing-actuation systems using transferable and reinforced learning," *IEEE Trans. Ind. Informat.*, vol. 19, no. 11, pp. 10811–10820, Nov. 2023.
[280] X. Xu, B. Shen, S. Ding, G. Srivastava, M. Bilal, M. R. Khosravi, V. G. Menon, M. A. Jan, and M. Wang, "Service offloading with deep Q-network for digital twinning-empowered Internet of Vehicles in edge computing," *IEEE Trans. Ind. Informat.*, vol. 18, no. 2, pp. 1414–1423, Feb. 2022.
[281] Z. Xu, J. Tang, J. Meng, W. Zhang, Y. Wang, C. H. Liu, and D. Yang, "Experience-driven networking: A deep reinforcement learning based approach," in *Proc. IEEE Conf. Comput. Commun.*, Apr. 2018, pp. 1871–1879.
[282] G. Yan, T. Zhou, B. Hu, Z.-Q. Fu, and B.-H. Wang, "Efficient routing on complex networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 73, no. 4, pp. 1–12, Apr. 2006, doi: 10.1103/physreve.73.046108.
[283] S. Yan, X. Wang, X. Zheng, Y. Xia, D. Liu, and W. Deng, "ACC: Automatic ECN tuning for high-speed datacenter networks," in *Proc. ACM SIGCOMM Conf.*, Aug. 2021, pp. 384–397.
[284] C. Yang, P. Liang, L. Fu, G. Cui, F. Huang, F. Teng, and Y. A. Bangash, "Using 5G in smart cities: A systematic mapping study," *Intell. Syst. Appl.*, vol. 14, May 2022, Art. no. 200065.
[285] H. Yang, Y. Li, K. Yao, T. Sun, and C. Zhou, "A systematic network traffic emulation framework for digital twin network," in *Proc. IEEE 1st Int. Conf. Digit. Twins Parallel Intell. (DTPI)*, Jul. 2021, pp. 94–97.
[286] J. Yang, F. Lin, C. Chakraborty, K. Yu, Z. Guo, A.-T. Nguyen, and J. J. P. C. Rodrigues, "A parallel intelligence-driven resource scheduling scheme for digital twins-based intelligent vehicular systems," *IEEE Trans. Intell. Vehicles*, vol. 8, no. 4, pp. 2770–2785, Apr. 2023.
[287] I. Yaqoob, K. Salah, M. Uddin, R. Jayaraman, M. Omar, and M. Imran, "Blockchain for digital twins: Recent advances and future research challenges," *IEEE Netw.*, vol. 34, no. 5, pp. 290–298, Sep. 2020.
[288] M. Yaqoob, R. Trestian, M. Tatipamula, and H. X. Nguyen, "Digital-twin-driven end-to-end network slicing toward 6G," *IEEE Internet Comput.*, vol. 28, no. 2, pp. 47–55, Mar. 2024.
[289] A. Yousefpour, C. Fung, T. Nguyen, K. Kadiyala, F. Jalali, A. Niakanlahiji, J. Kong, and J. P. Jue, "All one needs to know about fog computing and related edge computing paradigms: A complete survey," *J. Syst. Archit.*, vol. 98, pp. 289–330, Sep. 2019.
[290] K. Yu, L. Lin, M. Alazab, L. Tan, and B. Gu, "Deep learning-based traffic safety solution for a mixture of autonomous and manual vehicles in a 5G-enabled intelligent transportation system," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 7, pp. 4337–4347, Jul. 2021.
[291] M. Zalat, C. Barber, D. Krauss, B. Esfandiari, and T. Kunz, "Network routing optimization using digital twins," in *Proc. PoEM Companion*, 2023, pp. 1–8.
[292] Q. Zeng, Q. Sun, G. Chen, H. Duan, C. Li, and G. Song, "Traffic prediction of wireless cellular networks based on deep transfer learning and cross-domain data," *IEEE Access*, vol. 8, pp. 172387–172397, 2020, doi: 10.1109/ACCESS.2020.3025210.
[293] X. Zeng, R. Bagrodia, and M. Gerla, "GloMoSim: A library for parallel simulation of large-scale wireless networks," in *Proc. 12th Workshop Parallel Distrib. Simulation*, 1998, pp. 154–161.
[294] C. Zhang, Y. Xie, H. Bai, B. Yu, W. Li, and Y. Gao, "A survey on federated learning," *Knowl.-Based Syst.*, vol. 216, Jan. 2021, Art. no. 106775.
[295] H. Zhang, Q. Liu, X. Chen, D. Zhang, and J. Leng, "A digital twin-based approach for designing and multi-objective optimization of hollow glass production line," *IEEE Access*, vol. 5, pp. 26901–26911, 2017.
[296] M. Zhang, M. Polese, M. Mezzavilla, J. Zhu, S. Rangan, S. Panwar, and M. Zorzi, "Will TCP work in mmWave 5G cellular networks?" *IEEE Commun. Mag.*, vol. 57, no. 1, pp. 65–71, Jan. 2019.
[297] Z. Zhang, Y. Zhou, L. Teng, W. Sun, C. Li, X. Min, X.-P. Zhang, and G. Zhai, "Quality-of-experience evaluation for digital twins in 6G network environments," *IEEE Trans. Broadcast.*, vol. 70, no. 3, pp. 995–1007, Sep. 2024.
[298] L. Zhao, Z. Bi, A. Hawbani, K. Yu, Y. Zhang, and M. Guizani, "ELITE: An intelligent digital twin-based hierarchical routing scheme for softwarized vehicular networks," *IEEE Trans. Mobile Comput.*, vol. 22, no. 9, pp. 5231–5247, Sep. 2022.
[299] L. Zhao, G. Han, Z. Li, and L. Shu, "Intelligent digital twin-based software-defined vehicular networks," *IEEE Netw.*, vol. 34, no. 5, pp. 178–184, Sep. 2020.
[300] C. Zhou. (Jul. 2024). *Network Digital Twin: Concepts and Reference Architecture*. Internet Engineering Task Force.[Online]. Available: https://datatracker.ietf.org/doc/draft-irtf-nmrg-network-digital-twin-arch/06/
[301] X. Zhou, X. Zheng, X. Cui, J. Shi, W. Liang, Z. Yan, L. T. Yang, S. Shimizu, and K. I.-K. Wang, "Digital twin enhanced federated reinforcement learning with lightweight knowledge distillation in mobile networks," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 10, pp. 3191–3211, Oct. 2023.
[302] H. Zhu, V. Gupta, S. S. Ahuja, Y. Tian, Y. Zhang, and X. Jin, "Network planning with deep reinforcement learning," in *Proc. ACM SIGCOMM Conf.*, Aug. 2021, pp. 258–271.
[303] E. Zio and L. Miqueles, "Digital twins in safety analysis, risk assessment and emergency management," *Rel. Eng. Syst. Saf.*, vol. 246, Jun. 2024, Art. no. 110040.

REZA POORZARE (Member, IEEE) received the B.S. and M.S. degrees in computer engineering from the Azad University of Iran, in 2010 and 2014, respectively, and the Ph.D. degree in network engineering from the Universitat Politécnica de Catalunya, Barcelona, Spain, in 2022. After that, he was a Postdoctoral Researcher with the Institute of Data-Centric Software Systems (IDSS), Karlsruhe University of Applied Sciences, Karlsruhe, Germany, until April 2024, where he is currently a Research Associate. His research interests include 5G, TCP, MPTCP, congestion control, SDN, digital twin, and artificial intelligence.

DIMITRIS N. KANELLOPOULOS received the Diploma, M.Sc., and Ph.D. degrees from the Department of Electrical and Computer Engineering, University of Patras, Greece. Since 1990, he has been a Research Assistant with the Department of Electrical and Computer Engineering, University of Patras, and involved in several EU research and development projects. He was an Adjunct Faculty Member of the Technological Educational Institute of Patras. He is a member of the ESD Laboratory, Department of Mathematics, University of Patras. His research interests include multimedia networks, wireless ad-hoc networks, MPTCP, congestion control, the IoT, and digital twins. He is an Associate Editor of the journal *Human-centric Computing and Information Sciences*. He is an Editor for the Section ''Electronic Multimedia'' in *Electronics* (MDPI) and a Topic Editor of *Sensors* (MDPI) and *Electronics*. He serves as a Guest Editor for *Frontiers in Computer Science* and the MDPI journals *Electronics*, *Applied Sciences*, and *Information*. He is a Steering Committee Member of the International Conference on Emerging Networks and Systems Intelligence (EMERGING). He has served as a program committee member for more than 100 conferences. He serves as a reviewer for various IEEE, Springer, and Elsevier journals. He has edited two books on ''multimedia networking'' and has co-authored over 100 journal articles and conference papers (H-index: 26, i10-index: 54).

VARUN KUMAR SHARMA received the Ph.D. degree in computer science and engineering (network communication) from the Jaypee University of Engineering and Technology, Guna, India, in 2018. Currently, he is an Assistant Professor with the Department of Computer Science and Engineering, The LNM Institute of Information Technology, Jaipur, Rajasthan, India. His research interests include TCP, MPTCP, multipath schedulers, multipath/singlepath congestion control, performance of network and transport layer protocols in mobile ad-hoc networks, digital twin, wireless sensor networks, and the IoT.

POULAMI DALAPATI (Member, IEEE) received the B.Tech. degree in computer science and engineering from WBUT Salt Lake, Kolkata, India, in 2011, the M.Tech. degree in computer science from the Birla Institute of Technology Mesra, Ranchi, India, in 2013, and the Ph.D. degree in computer science and engineering (artificial intelligence) from the National Institute of Technology Durgapur, India, in 2018. She is currently an Assistant Professor with the Computer Science and Engineering Department, The LNM Institute of Information Technology, Jaipur, India. Her research interests include intelligent transportation systems, the Internet of Vehicles, digital twin, autonomous agent-based systems, and machine learning.

OLIVER P. WALDHORST (Member, IEEE) received the Diploma and Ph.D. degrees in computer science from the University of Dortmund, in 2000 and 2005, respectively, and the Habilitation degree from KIT, in 2011. Since September 2016, he has been a Professor in computer and communication networks with Karlsruhe University of Applied Sciences. His research interests include communication protocols, distributed applications in mobile environments, and performance analysis of communication systems. From 2000 to 2005, he was a Research Assistant with the University of Dortmund, followed by a postdoctoral role with the University of Leipzig, in 2006. Between 2006 and 2013, he was a Senior Researcher and a Lecturer with KIT, also working as a Visiting Researcher with the University of Toronto (2009–2010), and a Visiting Professor with TU Ilmenau (2011–2012). From 2013 to 2016, he contributed to connected car communication architecture development with Daimler AG.