SDN-Enabled Digital Twins: A Framework for Wireless SDN Simulation and Optimization

Published Version:
Trotta, A., Micciché, M., Heideker, A., Di Felice, M. (2025). SDN-Enabled Digital Twins: A Framework for Wireless SDN Simulation and Optimization. 345 E 47TH ST, NEW YORK, NY 10017 USA : Institute of Electrical and Electronics Engineers Inc. [10.1109/ccnc54725.2025.10976005].

Availability:
This version is available at: https://hdl.handle.net/11585/1037109 since: 2026-01-14

Published:
DOI: http://doi.org/10.1109/ccnc54725.2025.10976005

(Article begins on next page)

SDN-Enabled Digital Twins: A Framework for Wireless SDN Simulation and Optimization

Angelo Trotta*, Mario Miccichè*, Alexandre Heideker*, Marco Di Felice*†
* *Department of Computer Science and Engineering, University of Bologna, Italy*
† *Advanced Research Center on Electronic Systems "Ercole De Castro", University of Bologna, Italy*
Emails: {angelo.trotta5, alexandre.heideker, marco.difelice3}@unibo.it, mario.micciche@studio.unibo.it

***Abstract*—The dynamic nature of modern wireless networks, combined with the need for efficient management, has heightened interest in Software-Defined Networking (SDN). This paper presents a simulation framework designed to effectively model SDN in wireless network environments. The framework addresses key challenges in managing heterogeneous networks, offering a versatile platform for the development and evaluation of SDN applications. A key feature of the framework is the integration of a Digital Twin (DT) module within the SDN controller, leveraging the controller's comprehensive view of the network. This integration allows for real-time construction and updating of a virtual representation of the network, enhancing the controller's decision-making capabilities. By predicting future network states through a neural network model, the DT facilitates proactive management strategies, such as routing adjustments and resource reallocation, which are essential for maintaining optimal network performance. The paper details the architectural design and implementation of the framework, including the integration of Mininet, OMNeT++, and the Ryu controller. Our results demonstrate the framework's effectiveness in simulating complex SDN scenarios and providing detailed analyses of network behavior.**

***Index Terms*—SDN, Wireless Networks, Digital Twin, Simulation Framework, Mininet, Ryu, OMNeT++, Neural Networks**

I. Introduction

In recent years, Software-Defined Networking (SDN) has revolutionized the management of modern wireless networks. By decoupling the control plane from the data plane, SDN provides a centralized and programmable approach to network management, leading to enhanced flexibility, scalability, and efficiency. These requirements are especially crucial for wireless networks, where dynamic and efficient resource management is essential [1]. Such capabilities become even more relevant in scenarios involving multi-layered networks, such as those integrating satellite, aerial, and terrestrial segments. SDN's centralized control can provide real-time coordination across diverse network layers [2]. The concept of Digital Twins (DT) is also gaining significant traction in various industries, including network management. A DT is a virtual model of a physical system that allows for real-time monitoring, predictive maintenance, and optimization. For wireless networks, DTs can offer comprehensive insights into network operations, enabling proactive issue resolution and more effective resource utilization [3], [4]. In multi-layer network scenarios, a DT can further enhance this coordination, ensuring seamless connectivity and optimal resource distribution across various network segments. SDN's centralized control and programmability make it an ideal candidate for creating DTs of wireless networks. By continuously monitoring network performance and leveraging real-time data, SDNs can provide the foundation for accurate and dynamic DTs. This combination transforms network management, making it more responsive, predictive, and efficient [5], [6]. Despite these promising developments, current simulation tools face significant challenges in accurately simulating SDNs in wireless environments. Many existing tools are not adequately integrated, limiting their ability to provide comprehensive and realistic simulations [7]. These shortcomings hinder the effective simulation of SDNs, which is critical for developing and testing new network solutions.

This paper introduces a novel approach that integrates a DT into an SDN controller, leveraging the controller's comprehensive overview of the network to enable real-time construction and updating of the DT. The DT utilizes a neural network model that takes the history of network topology and node mobility as input to forecast future network states, such as changes in connectivity. This predictive capability allows for more advanced network management, enabling the SDN controller to make informed decisions, such as preemptively adjusting routing paths and reallocating resources to optimize network performance and mitigate potential issues before they arise. To support the study and evaluation of this system in a wireless environment, we propose a comprehensive simulation framework that combines Mininet[1], Ryu[2], and OMNeT++[3]. Mininet, a network emulator, provides a virtual environment to model SDN-enabled networks, while OMNeT++, a discrete event simulator, accurately simulates wireless network behavior, including node mobility and wireless link dynamics. Ryu, a flexible SDN controller platform, manages the network's control plane, enabling the simulation of the interaction between the SDN controller, switches, and wireless nodes. Together, these tools provide a robust platform for exploring the impact of the DT on network performance. Key contributions of this paper include:

- The proposal and implementation of a DT within an SDN controller, enabled by the controller's centralized view of the network. The DT is built using a neural network ar-

[1] http://mininet.org/
[2] https://ryu-sdn.org/
[3] https://omnetpp.org/

chitecture that combines Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) layers to predict the connectivity of the wireless network, allowing for real-time monitoring and predictive decision-making.
- The design of a simulation framework that integrates Mininet, Ryu, and OMNeT++ to model and study the interaction between the DT and wireless network environments, providing insights into the dynamic behavior of SDN-enabled wireless networks.
- The evaluation of the framework's performance, including the extraction and analysis of key wireless network metrics such as delay and SDN control overhead. Additionally, the performance of the DT is assessed by evaluating its prediction accuracy and its impact on network management and optimization.

Our results demonstrate that the proposed framework provides accurate simulations of SDN-enabled wireless environments. Additionally, the incorporation of a CNN-LSTM-based DT within the SDN controller allows for real-time monitoring and proactive adjustments, resulting in better resource allocation and reduced network disruptions.

The rest of the paper is structured as follows. Section II reviews existing simulation tools for SDN and wireless networks and discusses the application of DT in network management. Section III presents the system model. Section IV discusses the DT model, including its neural network architecture and predictive capabilities. Section V details the implementation of the framework, highlighting the integration of Mininet, Ryu, and OMNeT++. Section VI validates the proposed framework and evaluates the effectiveness of the DT in optimizing network operations. Section VII concludes the work.

II. Related Work

This section explores the existing work relevant to our proposed framework by reviewing simulation tools used in SDN and wireless networks and the application of DTs in network management. The objective is to outline the tools and technologies that inform our framework and the specific advancements our approach offers in the context of DT applications.

Several tools have been developed to simulate SDNs and wireless networks. Mininet-WiFi, introduced by [8], allows for the emulation of software-defined wireless networks, providing a controlled environment for testing SDN solutions. Similarly, [9] presents SDN-Sim, a system-level simulator that integrates with SDN to enable detailed network simulations and performance analysis. OpenNet, developed by [10], facilitates the study of SDN-based network management in wireless local area networks. Additionally, SDN-WISE by [11] offers a stateful SDN solution for wireless sensor networks, enhancing network efficiency and scalability. The combination of Mininet and ns-3, as discussed by [12], addresses the challenges of emulating SDN environments. Furthermore, [13] compares various SDN simulators, including Mininet and OPNET, providing insights into their strengths and limitations. Lastly, [14] conducts a benchmarking analysis of SDN emulators, evaluating their performance and suitability for different types of network research. In addition to these tools, [15] introduces a flexible simulation framework specifically for distributed SDN-controller architectures using OMNeT++. This framework allows for evaluating the performance of distributed SDN controllers, such as Hyperflow and Kandoo, and explores the trade-offs between resiliency, synchronization, and scalability in different controller architectures.

DTs have been increasingly applied in network management, particularly in wireless networks. [3] introduces the Relativistic DT (RDT) framework, which addresses the challenge of deploying DTs in heterogeneous IoT environments. The RDT framework automatically generates general-purpose DTs across various IoT entities, regardless of the data formats and network protocols in use. The relevance of DTs in SDN environments is discussed in [4], highlighting their potential to enhance network resilience and performance. Moreover, [6] emphasizes the role of DTs in the industrial Internet of Things (IIoT), showcasing their benefits in predictive maintenance and real-time monitoring. [16] explores the use of DTs in software-defined UAV networks, utilizing queuing models to enhance network performance and reliability. The application of neural networks in network monitoring and prediction is also gaining attention. [17] presents AMoND, an area-controlled mobile ad-hoc networking framework with DT technology, aiming to improve network management and efficiency. [18] provides a comprehensive overview of DT technology in wireless systems, discussing the taxonomy, challenges, and opportunities. Furthermore, [19] delves into the construction of DT networks using SDN and knowledge graphs, highlighting their potential benefits for network management and optimization.

These studies underscore the potential of DT and advanced simulation tools to improve network management and optimization. Our proposed framework builds on these insights by integrating Mininet, Ryu, and OMNeT++ to create a unified simulation environment for wireless networks. A key innovation of our framework is the utilization of the SDN controller's centralized view of the network to construct and maintain an accurate DT. This centralized knowledge base enables the real-time creation and updating of the DT, which provides detailed and predictive insights into network states. This integration not only facilitates the accurate simulation of SDN controllers and switches in wireless settings but also significantly enhances real-time network monitoring, predictive maintenance, and overall network management efficiency.

III. System Model

The proliferation of wireless devices, including sensors, drones, and other IoT devices, demands advanced network management solutions capable of handling dynamic and complex environments, particularly those involving multi-layered architectures such as satellite, aerial, and ground segments. SDN offers a centralized and programmable approach to network management by decoupling the control plane from the data plane. To enhance SDN's capabilities further, we propose integrating a DT module into the SDN controller, which provides real-time monitoring and predictive analytics to optimize network operations.

The proposed framework features a layered architecture that integrates wireless nodes equipped with SDN switches, an SDN controller, and a DT module.

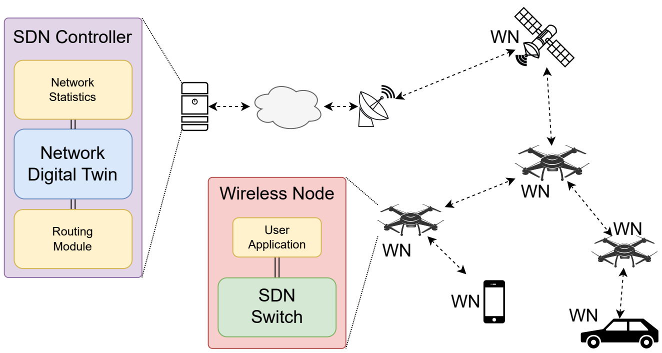


Fig. 1: Conceptual architecture of the proposed framework.

The high-level architecture, illustrated in Figure 1, demonstrates an application in a Non-Terrestrial Network (NTN) and is composed of the following components:

- **Wireless Nodes (WN)**: these encompass various devices such as IoT sensors, mobile nodes, satellites, and drones. Each node has a layered architecture, with an SDN switch at the bottom layer enabling programmable control over network traffic and resource allocation. The user application layer resides above the switch, running node-specific applications or services. This separation between data handling and application functions allows for more efficient and flexible network management.
- **SDN Controller**: acting as the central hub for network management, the SDN controller maintains a global view of the network state, informed by real-time data from SDN switches. The DT module, integrated within the controller, uses this data from the *network statistics* module to maintain an up-to-date virtual representation of the network. By predicting future conditions, such as congestion or link failures, the DT provides optimization suggestions to the *routing module*, enabling the controller to make proactive adjustments that improve network performance and stability.

Although Figure 1 illustrates a use case involving satellite, drone, and ground users typical of non-terrestrial networks, the system model is adaptable to various wireless network environments, providing flexibility for a wide range of applications.

The integration of the DT within the SDN framework establishes a feedback loop for proactive network management. Data collected from wireless nodes is processed by the DT, which informs the SDN controller's decisions on routing, load balancing, and resource allocation. This process enhances network efficiency and reliability by anticipating and mitigating potential issues before they affect performance.

IV. Proposed Digital Twin Model

The DT model in our simulation framework enhances the SDN controller by providing predictive insights into future network states, allowing for proactive network management. By leveraging real-time data, such as network topology and traffic patterns, the DT forecasts conditions like congestion or node failures, enabling the SDN controller to optimize routing and resource allocation. This section details the neural network architecture of the DT and presents use cases demonstrating its impact on network performance optimization. The core of the DT's predictive capabilities is a deep neural network designed to process multiple inputs related to the network's state and forecast future connectivity matrices. The architecture of this neural network is illustrated in Figure 2.

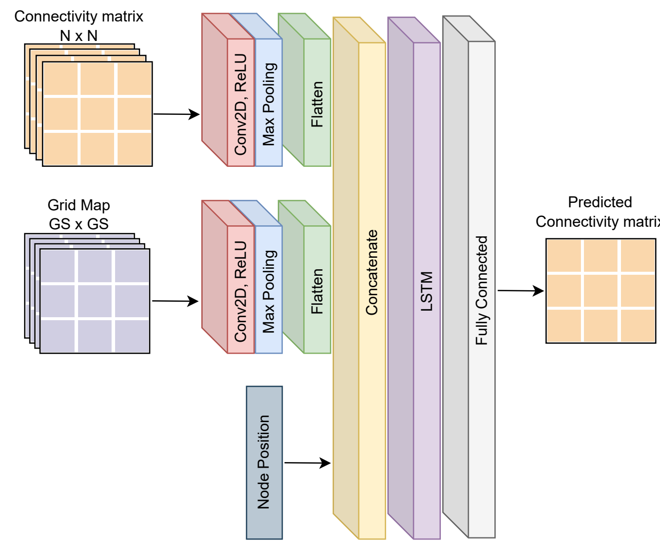


Fig. 2: Topology of the neural network used in the DT model. The architecture includes convolutional layers for matrix inputs and LSTM layers for sequence processing.

The neural network processes three distinct inputs: the network's connectivity matrix, a grid-map of the nodes' spatial distribution, and a list of node positions. The model predicts future connectivity matrices based on these inputs. Here, we assume that the SDN controller manages a map of size $S \times S$ square meters and N wireless switches. The architecture comprises the following components:

- The model takes three inputs: (i) a connectivity matrix of size $N \times N$ representing the network's connections at each time step; (ii) a grid-map of size $GS \times GS$, where $GS = \lceil S/C \rceil$ with a cell-size granularity of C meters, capturing the spatial arrangement of the nodes; and (iii) a list of the N node positions, each represented by its x and y coordinates.
- The connectivity matrix and grid-map are processed by separate convolutional neural networks (CNNs). Each matrix is passed through a series of 2D convolution layers followed by max pooling to extract spatial features. The CNN outputs are then flattened for further processing.

- The outputs from the CNNs (for both the connectivity matrix and grid-map) are concatenated with the node position data. This combined representation integrates the connectivity structure, spatial layout, and node-specific information into a single feature vector.
- After the merging step, the feature vector is passed through Long Short-Term Memory (LSTM) layers to capture temporal dependencies and forecast future network states. These layers process the time series data to predict how the network's connectivity will evolve.
- The output is a time-distributed dense layer with a sigmoid activation function, which predicts the future connectivity matrix for the next forecasted time step.

This architecture allows the DT to accurately predict future connectivity matrices based on current network conditions, including the spatial distribution of nodes and their positions over time. The use of convolutional layers for matrix inputs, combined with LSTM layers for sequence forecasting, enables the model to capture both spatial and temporal patterns in the network. The rationale behind this architecture is to potentially include matrices related to wireless hop-by-hop metrics such as RSSI, PDR, delay, and throughput in future work, further enriching the capabilities and decision-making power of the SDN controller.

V. Simulation Framework for SDN-Enabled Optimization of Wireless Networks

In wireless networks, analyzing real-world scenarios can be challenging due to the difficulty of deploying and managing large numbers of wireless devices, particularly in complex or dynamic environments. Studying the scalability and behavior of wireless networks in such scenarios is often impractical without a controlled and flexible testing environment. To address these challenges, simulation frameworks play a crucial role in enabling researchers to evaluate network performance and scalability under various conditions without requiring extensive physical infrastructure. The simulation framework developed for this study integrates Mininet, OMNeT++, and Ryu to provide a robust environment for simulating and optimizing wireless networks. This framework allows for the simulation of large-scale wireless network topologies and enables the collection of detailed network state information, including positional data, which enhances the SDN controller's ability to manage the network proactively. This section details the framework's architecture and implementation, highlighting the integration of these tools and the use of custom SDN messages to enhance network monitoring and management. The architecture of the proposed simulation framework is depicted in Figure 3. The framework integrates three primary components: Mininet, OMNeT++, and the Ryu SDN controller.

- **Mininet** serves as the emulation environment for the network's virtualized infrastructure, creating a fully connected mesh topology of SDN-enabled nodes. Each node consists of a host that contains the application logic and an SDN switch. In this topology, all nodes are connected through virtual links, abstracting the underlying physical connections. This abstraction allows the SDN controller to manage the network as if it were fully wired, while OMNeT++ simulates the actual wireless links and their behavior.

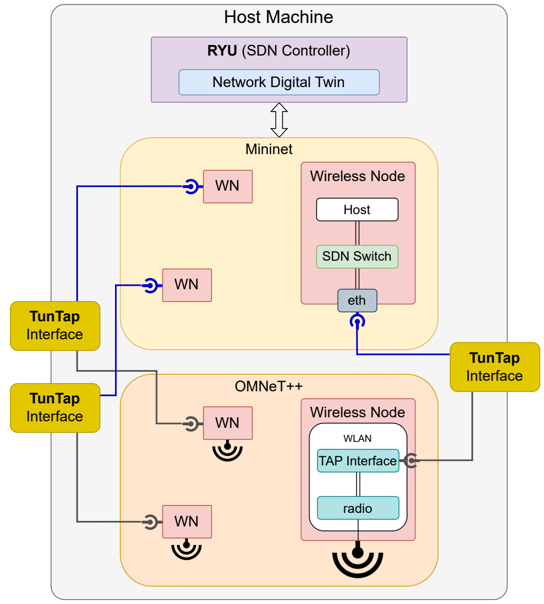


Fig. 3: Architecture of the simulation framework.

- **OMNeT++** simulates the wireless aspects of the network, including the dynamic nature of wireless communication channels, signal propagation, and node mobility. While Mininet provides the virtual links in a fully connected topology, OMNeT++ characterizes these links by simulating realistic wireless conditions, such as varying signal strengths, interference, and data rates. This setup ensures that the simulated network conditions closely resemble real-world wireless networks.
- The **Ryu SDN controller** acts as the network's central management system, utilizing its global view to enforce network policies, manage traffic flows, and allocate resources. The controller communicates with the SDN switches in Mininet via the OpenFlow protocol, directing how data should be forwarded across the network based on the latest network state information.

An important feature of this framework is that it is built on the standard OpenFlow protocol, ensuring compatibility with both simulated and real-world network environments. This means that, beyond its simulation capabilities, the framework could be deployed in real-world environments with minimal modification. The Ryu controller in this setup interacts with both virtual and real wireless nodes using standard OpenFlow messages, allowing it to manage real network devices in the same way as it does in the simulation. This ensures a straightforward transition from simulation to real-world implementation.

In addition to standard OpenFlow messages, our framework extends the OpenFlow protocol with custom SDN messages. These custom messages are designed to collect additional information from SDN-enabled switches, such as the geographic positions of wireless nodes, which is not part of the standard OpenFlow message set. By embedding these custom fields within the OpenFlow framework, the system enhances its ability to monitor and manage network conditions, while still maintaining compliance with the core OpenFlow protocol. The custom message workflow includes the following three steps. (i) *Message creation*: the SDN controller sends out custom SDN messages with empty fields to the wireless nodes, requesting specific data such as node positions. (ii) *Data filling*: as these messages are routed through the network, they pass through OMNeT++. OMNeT++ has detailed simulation data, including the exact positions of nodes. It intercepts these messages and fills empty fields with the requested data, such as node positions. (iii) *Data integration*: the enriched messages, now containing the actual data, continue to the SDN controller. The controller then integrates this information into its network state database, improving its understanding of network topology and node dynamics. This approach allows the SDN controller to maintain a dynamic and accurate model of the network, which is crucial for optimizing network performance, particularly in scenarios involving mobility and varying wireless link conditions.

Implementation Details

The implementation of the proposed simulation framework required the use of specific tools and techniques to ensure seamless interaction between the different components. Two particular aspects of the implementation are the use of TUN/TAP interfaces for communication between Mininet and OMNeT++ and the external deployment of the Ryu SDN controller. These choices enable efficient message handling and flexible simulation of wireless environments.

1) TUN/TAP Interface for Communication: To facilitate communication between Mininet and OMNeT++, we employed TUN/TAP virtual network interfaces. These interfaces enable network traffic redirection, allowing Mininet to interact seamlessly with OMNeT++. The TUN/TAP devices capture packets from Mininet's virtual switches and forward them to OMNeT++ to simulate wireless transmission characteristics. On the other hand, OMNeT++ can inject packets inside Mininet, completing the bidirectional flow of data.

2) External Ryu Controller Deployment: The Ryu SDN controller is deployed on an external server in the proposed framework. This design choice simplifies the setup and allows the controller to handle control messages via direct socket connections, independent of the simulated wireless environment in OMNeT++. While this separation streamlines the implementation, it does not fully reflect real-world scenarios where control messages might traverse the same wireless links as data traffic. In future work, we plan to incorporate the transmission of control messages through the simulated wireless environment. This enhancement will enable the differentiation between data and control planes, providing more accurate modeling of network behaviors and allowing for the exploration of different quality-of-service strategies for control and data traffic. The described implementation details not only support the current functionality of the framework but also lay the groundwork for future enhancements that will bring the simulation closer to real-world network operations.

VI. Evaluation and Results

In this Section, we evaluate the effectiveness of the proposed simulation framework and the integrated DT model. The evaluation is divided into two parts: first, we validate the framework by analyzing key network performance metrics such as delay and SDN control overhead, obtained through simulations of various wireless network scenarios. Second, we assess the performance of the DT model by evaluating its predictive accuracy in forecasting network states and its impact on network management decisions.

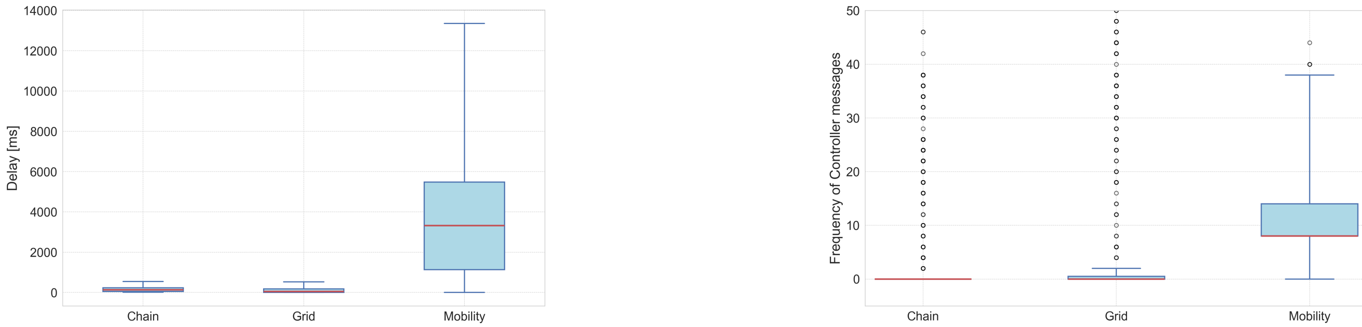
A. Validation of the Simulation Framework

To validate the proposed simulation framework and its capability of capturing essential features of an SDN-enabled wireless network, we deployed a set of four wireless nodes in various topologies, including both static and dynamic mobility scenarios. The static topologies considered were a linear *chain* configuration and a 2×2 *grid* configuration. Additionally, we introduced a dynamic *mobility* scenario where the nodes randomly moved within a predefined area of 500×500 meters. In these experiments, the DT's predictive capabilities for link failures were not active, allowing for an evaluation focused solely on the framework's baseline performance.

The evaluation focused on two key performance metrics:

- **End-to-End Delay**: This metric measures the time taken for a data packet to travel from the source node to the destination node across the network. It provides insight into the efficiency and responsiveness of the network under different topological and mobility conditions.
- **Frequency of SDN Control Messages**: This metric quantifies the amount of control traffic generated by the SDN controller to manage the network. It is a critical factor in understanding the scalability of the network and the effectiveness of the SDN controller in handling various network scenarios.

Figure 4a presents the average delay experienced by data packets in each topology. The static chain topology typically exhibits lower delays due to the stable and consistent paths between nodes, while the 2×2 grid configuration also maintains relatively low delays. In contrast, the dynamic mobility scenario results in significantly higher delays as nodes move, causing frequent path breaks and necessitating continuous route recalculations by the SDN controller. Figure 4b illustrates the overhead generated by the SDN controller in each scenario. The static topologies show relatively stable overhead levels, with the grid generating slightly more control messages due to occasional packet losses on the diagonal links. However, the dynamic mobility scenario results in a substantially higher overhead, as the controller must frequently



(a) (b)

Fig. 4: Average end-to-end delay and SDN control message overhead for different network topologies and mobility scenarios, respectively in Figure 4a and Figure 4b.

adapt to the changing network conditions, issuing more control messages to re-establish routes and maintain network connectivity.

B. Digital Twin Performance Evaluation

To assess the performance of the DT model, we utilized synthetic data generated by the proposed simulation framework. The SDN controller was responsible for collecting all the required data, including connectivity matrices and node positions. This synthetic dataset allowed for a comprehensive evaluation of the DT's predictive capabilities under various network conditions. We conducted simulations with different numbers of wireless switches equipped with an 802.11b network interface, N, ranging from 5 to 50, with random mass-mobility within a scenario of size $S \times S$. For the experiments, we set $S = 1$km and a cell size $C = 10$m. We compared the performance of our proposed neural network architecture, which includes convolutional layers for spatial inputs, against a baseline LSTM-only model with a comparable number of learning parameters. The evaluation focused on two key metrics: the F1-score for connectivity matrix prediction and the F1-score related to accurate routing path predictions. Both metrics were evaluated by varying the number of wireless nodes in the network.

1) Connectivity Matrix Prediction: The first evaluation metric is the F1-score for predicting the future connectivity matrix. This metric measures how accurately the DT model can forecast the network's future connections based on current inputs. Figure 5a shows that both models perform well, with F1-scores remaining high across varying network sizes. For networks with a small number of nodes, the LSTM-only model is similar or slightly outperforms our proposed model. However, as the number of nodes increases, our proposed model consistently matches or exceeds the performance of the LSTM-only model, demonstrating its ability to handle more complex spatial-temporal data.

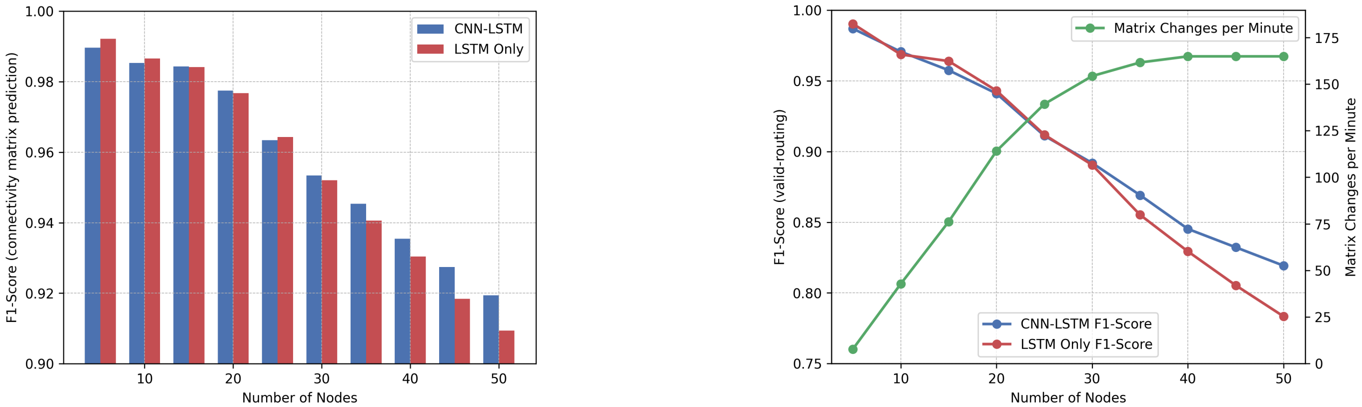
2) Routing Path Prediction and Network Stability: The second evaluation focuses on how well the DT model predicts routing paths that can be maintained or adjusted based on accurate predictions of the network's future state. We define a prediction as correct when either (i) the actual routing path is broken, and the DT accurately predicts the necessary path change, or (ii) there is no break in the routing path, and the DT correctly predicts that no change is required. This metric is crucial for understanding the DT's effectiveness in preventing unnecessary route recalculations, which can lead to network instability and performance degradation. Figure 5b illustrates the F1-score for routing path predictions across different network sizes. As the number of nodes increases, the network topologies undergo more frequent changes, which negatively affects the performance of both models. Despite this, our proposed model demonstrates superior performance, especially in larger networks. For a small number of nodes, the LSTM-only model again is similar or slightly outperforms, but the gap closes as the node count increases, and our model consistently handles the increased complexity better. This result highlights the importance of incorporating spatial data through convolutional layers, which enhances the DT's ability to predict routing paths more accurately in more dynamic environments.

Overall, these results demonstrate that our proposed neural network architecture, combining convolutional layers and LSTM layers, offers a robust solution for predicting both connectivity matrices and routing paths in increasingly complex wireless network environments. The performance gap between our model and the LSTM-only baseline increases as the number of nodes grows, further validating the benefits of incorporating spatial data into the model.

VII. Conclusions

This paper presented a novel simulation framework that integrates Mininet, Ryu, and OMNeT++ to enable the study and development of Software-Defined Networking (SDN) applications in wired and wireless environments. The framework provides a comprehensive platform for simulating complex network scenarios, making it especially suitable for heterogeneous networks. By combining SDN and Digital Twin (DT) technologies, the framework supports real-time monitoring and predictive network management, enhancing the controller's decision-making capabilities.

One of the framework's core innovations is the integration of a DT module within the SDN controller. This DT



(a) (b)

Fig. 5: F1-score for connectivity matrix prediction varying number of wireless nodes in Figure 5a. F1-score for routing path predictions varying the number of wireless nodes in Figure 5b.

leverages neural network-based predictions to anticipate future network conditions, allowing the SDN controller to implement proactive strategies such as preemptive routing and resource allocation. Our evaluation demonstrates that the DT module significantly reduces route recalculations and improves overall network stability, particularly in dynamic environments with changing topologies. Additionally, the framework's compatibility with standard OpenFlow protocols ensures it can transition from simulated to real-world environments with minimal modifications.

Future work will focus on refining the predictive capabilities of the DT, expanding the framework to support more advanced network scenarios, and exploring its deployment in real-world settings.

Acknowledgment

The work has been supported by the PRIN 2022 project RAIN4C - "Reliable Aerial and satellIte Networks: joint Communication, Computation, Caching for Critical scenarios" (Project id: 20227N3SPN, CUP: J53D23007020001).

References

[1] K. Wang, Y. Wang, D. Zeng, and S. Guo, "An sdn-based architecture for next-generation wireless networks," *IEEE Wireless Communications*, vol. 24, no. 1, pp. 25–31, 2017.

[2] A. Trotta, A. Heideker, I. Zyrianoff, G. Interdonato, and S. Pizzi, "Advancing Non-Terrestrial Networks for Critical Scenarios with the RAIN4C Framework," in *2024 20th International Conference on Distributed Computing in Smart Systems and the Internet of Things (DCOSS-IoT)*, 2024, pp. 780–782.

[3] L. Sciullo, A. De Marchi, A. Trotta, F. Montori, L. Bononi, and M. Di Felice, "Relativistic Digital Twin: Bringing the IoT to the future," *Future Generation Computer Systems*, vol. 153, pp. 521–536, 2024.

[4] A. Dvir, Y. Haddad, and A. Zilberman, "The controller placement problem for wireless sdn," *Wireless Networks*, vol. 25, pp. 4963–4978, 2019.

[5] S. S. A. Gilani, A. Qayyum, R. N. B. Rais, and M. Bano, "Sdnmesh: An sdn based routing architecture for wireless mesh networks," *IEEE Access*, vol. 8, pp. 136 769–136 781, 2020.

[6] M. Kherbache, M. Maimour, and E. Rondeau, "Network digital twin for the industrial internet of things," in *2022 IEEE 23rd International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, 2022, pp. 573–578.

[7] B. Cao, Y. Li, C. Wang, G. Feng, S. Qin, and Y. Zhou, "Resource allocation in software defined wireless networks," *IEEE Network*, vol. 31, no. 1, pp. 44–51, 2017.

[8] R. R. Fontes, S. Afzal, S. H. B. Brito, M. A. S. Santos, and C. E. Rothenberg, "Mininet-wifi: Emulating software-defined wireless networks," in *2015 11th International Conference on Network and Service Management (CNSM)*, 2015, pp. 384–389.

[9] S. Ghosh, S. A. Busari, T. Dagiuklas, M. Iqbal, R. Mumtaz, J. Gonzalez, S. Stavrou, and L. Kanaris, "Sdn-sim: Integrating a system-level simulator with a software defined network," *IEEE Communications Standards Magazine*, vol. 4, no. 1, pp. 18–25, 2020.

[10] M.-C. Chan, C. Chen, J.-X. Huang, T. Kuo, L.-H. Yen, and C.-C. Tseng, "Opennet: A simulator for software-defined wireless local area network," in *2014 IEEE Wireless Communications and Networking Conference (WCNC)*, 2014, pp. 3332–3336.

[11] L. Galluccio, S. Milardo, G. Morabito, and S. Palazzo, "Sdn-wise: Design, prototyping and experimentation of a stateful sdn solution for wireless sensor networks," in *2015 IEEE Conference on Computer Communications (INFOCOM)*, 2015, pp. 513–521.

[12] S. Khan, A. Akram, H. Alsaif, and M. Usman, "Emulating software defined network using mininet-ns3-wifi integration for wireless networks," *Wireless Personal Communications*, vol. 118, no. 1, pp. 75–92, 2021.

[13] S. Lee, J. Ali, and B.-h. Roh, "Performance comparison of software defined networking simulators for tactical network: Mininet vs. opnet," in *2019 International Conference on Computing, Networking and Communications (ICNC)*, 2019, pp. 197–202.

[14] M. Hasan, H. Dahshan, E. Abdelwanees, and A. Elmoghazy, "Sdn mininet emulator benchmarking and result analysis," in *2020 2nd Novel Intelligent and Leading Emerging Sciences Conference (NILES)*, 2020, pp. 355–360.

[15] N. Gray, T. Zinner, S. Gebert, and P. Tran-Gia, "Simulation framework for distributed sdn-controller architectures in omnet++," in *Mobile Networks and Management: 8th International Conference, MONAMI 2016, Abu Dhabi, United Arab Emirates, October 23-24, 2016, Proceedings 8*. Springer, 2017, pp. 3–18.

[16] M. A. B. S. Abir and M. Z. Chowdhury, "Digital twin-based software-defined uav networks using queuing model," in *2023 10th International Conference on Signal Processing and Integrated Networks (SPIN)*, 2023, pp. 479–483.

[17] S. Ono, T. Yamazaki, T. Miyoshi, A. Taya, Y. Nishiyama, and K. Sezaki, "Amond: Area-controlled mobile ad-hoc networking with digital twin," *IEEE Access*, vol. 11, pp. 85 224–85 236, 2023.

[18] L. U. Khan, Z. Han, W. Saad, E. Hossain, M. Guizani, and C. S. Hong, "Digital twin of wireless systems: Overview, taxonomy, challenges, and opportunities," *IEEE Communications Surveys & Tutorials*, vol. 24, no. 4, pp. 2230–2254, 2022.

[19] 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. 63 092–63 106, 2023.