Intelligent Data-Driven Architectural Features Orchestration for Network Slicing

Rodrigo Moreira, Flavio de Oliveira Silva, Tereza Cristina Melo de Brito Carvalho, Joberto Martins

Intelligent Data-Driven Architectural Features Orchestration for Network Slicing

RODRIGO MOREIRA**, Universidade Federal de Viçosa (UFV), Brazil
FLÁVIO DE OLIVEIRA SILVA, Universidade do Minho, Portugal
TEREZA CRISTINA MELO DE BRITO CARVALHO, Universidade de São Paulo (USP), Brasil
JOBERTO S. B. MARTINS, Universidade Salvador (UNIFACS), Brazil

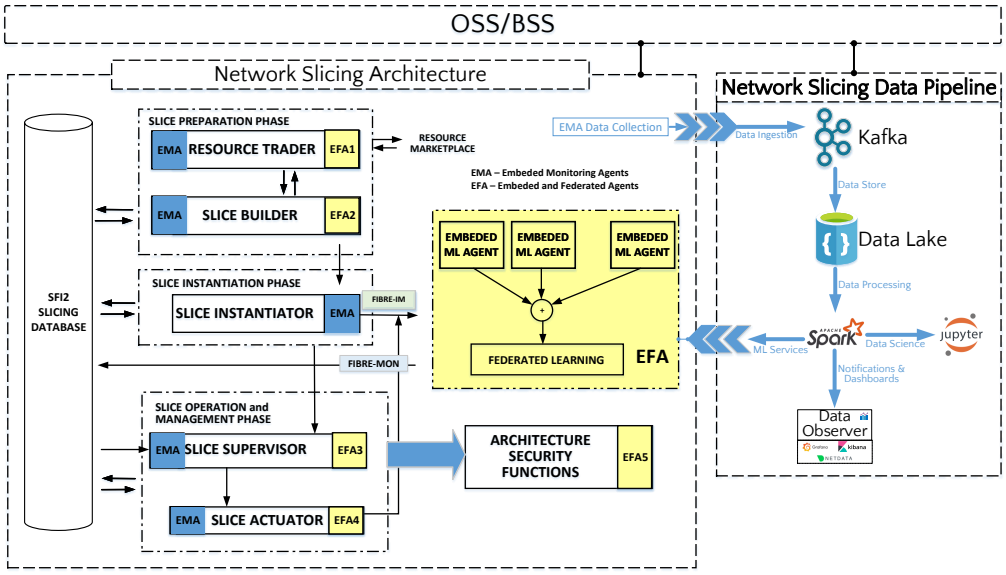

Fig. 1. Intelligent Data-Driven Architectural Features Orchestration for Network Slicing

Network slicing is a crucial enabler and a trend for the Next Generation Mobile Network (NGMN) and various other new systems like the Internet of Vehicles (IoV) and Industrial IoT (IIoT). Orchestration and machine learning are key elements with a crucial role in the network-slicing processes since the NS process needs to orchestrate resources and functionalities, and machine learning can potentially optimize the orchestration process. However, existing network-slicing architectures lack to define intelligent approaches to orchestrate features and resources in the slicing process. This paper discusses machine learning-based orchestration of features and capabilities in network slicing architectures. Initially, the slice resource orchestration and allocation in the slicing planning, configuration, commissioning, and operation phases are analyzed. In sequence, we highlight the need for optimized architectural feature orchestration and recommend using ML-embed agents, federated learning intrinsic mechanisms for knowledge acquisition, and

*All authors contributed equally to this research.

Manuscript submitted to ACM

a data-driven approach embedded in the network slicing architecture. We further develop an architectural features orchestration case embedded in the SFI2 network slicing architecture. An attack prevention security mechanism is developed for the SFI2 architecture using distributed embedded and cooperating ML agents. The case presented illustrates the architectural feature's orchestration process and benefits, highlighting its importance for the network slicing process.

CCS Concepts: • **Computing methodologies** → **Machine learning**; **Distributed artificial intelligence**; **Multi-agent systems**; **Cooperation and coordination**; • **Networks** → **Programmable networks**; *Network management*; **Network design principles**; *Network dynamics.*

Additional Key Words and Phrases: Network Slicing, Orchestration, Architectural Features, Intelligent Orchestration, ML-Native Slicing Architecture, Federated Learning, Data-Driven Slicing Architecture, SFI2, Security, ML-Native Security

ACM Reference Format:
Rodrigo Moreira, Flávio de Oliveira Silva, Tereza Cristina Melo de Brito Carvalho, and Joberto S. B. Martins. 2018. Intelligent Data-Driven Architectural Features Orchestration for Network Slicing. In *ADVANCE '24: Proceedings of the International Workshop on ADVANCEs in ICT Infrastructures and Services, February 26–29, 2024, Hanoi, Vietnam.* ACM, New York, NY, USA, 12 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Network slicing (NS) is a crucial enabler that supports virtual networks' planning, commissioning, configuration, operation, and management phases. Network slicing virtualizes physical resources like edge facilities, machines, communication links, switches, and radio access networks (RAN) while concomitantly allowing their customization and optimization [20] [2].

NS is adopted mainly in the next-generation mobile network (NGMN) (5G/6G) domain basically due to its inherent optimization capabilities that are necessary to accommodate the highly dynamic and variable requirements imposed by mobile users [8]. NS is also a trend in other domain areas such as Vehicular Networks [21], experimental networks [], and industrial IoT [22], among others, due to its capability to optimize and customize the network delivered for the user.

NS is an elaborated multi-phase process involving various architectural components. The existing network-slicing architectures like the ones proposed by 3GPP [1], IETF [9], ITU-T [10], ETSI [7], SFI2 project [11] and NECOS project [5] aim to structure the overall slicing process. The NS process is structured by proposing architectural components, segmenting, and sequencing activities like preparation, commissioning, operation, and decommissioning. Distinct network-slicing architectures adopt approximately the same architectural components and use equivalent sequencing for the slicing process. Some variations exist in their component's structure and features and phase interrelations. Another aspect that varies among NS architectures is the architecture customization concerning the target user (ISPs, mobile users, experimental networks, among others). Finally, a common agreement point among standardization bodies and researchers is that artificial intelligence and machine learning are integral parts of the solution, focusing more specifically on optimizations[15] [14].

However, although existing NS architectures have addressed NS phases and sequencing, solutions fall short of considering or recommending the orchestration strategy and approach among components, features, and resources. In this regard, machine learning utilization for NS optimization has to consider orchestration at various levels.

This paper addresses the issue of intelligent orchestration of features and resources towards a more efficient and robust network-slicing architecture. The proposed approach embeds intelligent agents in existing NS architecture components and phases to provide inherent services of various types and allow an improved architecture operation.

This paper is organized as follows. Section 1 introduces the network slicing current scenario. Section 2 presents the network slicing architectures and the architectural features orchestration in current NS architectures. Sections 3 and 4 present a set of architectural recommendations, followed by Section 5 presenting an architectural features orchestration case for security. Finally, Section 6 closes the discussion with the final considerations.

2 NETWORK SLICING ARCHITECTURES AND ORCHESTRATION

Network slicing architectures share a common group of functional components and phases like the ones proposed by the 3GPP NS initiative [1]. These common phases are:

- Preparation Phase: - In the preparation phase, the slice request is received and interpreted, and the necessary resources are identified and localized in the resource market. In this phase, orchestration occurs in terms of selecting resources from multiple domains or multiple options available on a single domain.
- Commissioning Phase - This phase consists basically of making choices among the available resources aiming to configure the requested slice service. Orchestration at this phase occurs by making configuration choices that can potentially optimize resource allocation among deployed slices for a slice provider.
- Operation Phase - In the operation phase, the slice is already deployed and operational. Orchestration may occur in this phase, mainly due to dynamic user traffic patterns at different slice deployments.
- Decommissioning Phase- In this phase, the allocated slice resources from single or multi-domains are liberated.

As indicated, orchestration may occur at different steps of the network slicing process and inherently involves multiple components of a network slicing architecture.

2.1 The SFI2 Network Slicing Reference Architecture

The SFI2 project (Slicing Future Internet Infrastructures) defines a network slicing reference architecture, named SFI2 architecture, that aims to integrate experimental networks, incorporating architectural advances like ML-native optimizations, energy-efficient slicing, and slicing-tailored security functionalities [11] (Figura 2).

In Figure 2, the SFI2 architecture is deployed for the experimental FIBRE[1] domain and allows its users to create virtual slices across the 18 islands of the FIBRE network with virtual machines, IoT resources, and communications links. The SFI2 functionalities and operation are explored in sequence, aiming to identify the architectural features that are the object of discussion in this paper.

In the SFI2 FIBRE deployment, the list and description of resources that can be allocated to create user slices are available through the marketplace functionality. The FIBRE marketplace stores the list of resource descriptions that interact with SFI2 architecture during the preparation phase and can include some trading activities between the SFI2 FIBRE deployment and the resource provider, in this case, the FIBRE domain.

The slice builder, as the name suggests, builds the requested user slice considering the resources obtained from the marketplace and may optimize the utilization of these resources in relation to the set of currently allocated resources used by the set of actively deployed slices.

The SFI2 slice instantiation deploys the configured slice in the FIBRE domain through a customized instantiation manager and sets up the required monitoring facilities for slice monitoring.

[1]FIBRE - Future Internet Brazilian Environment for Experimentation

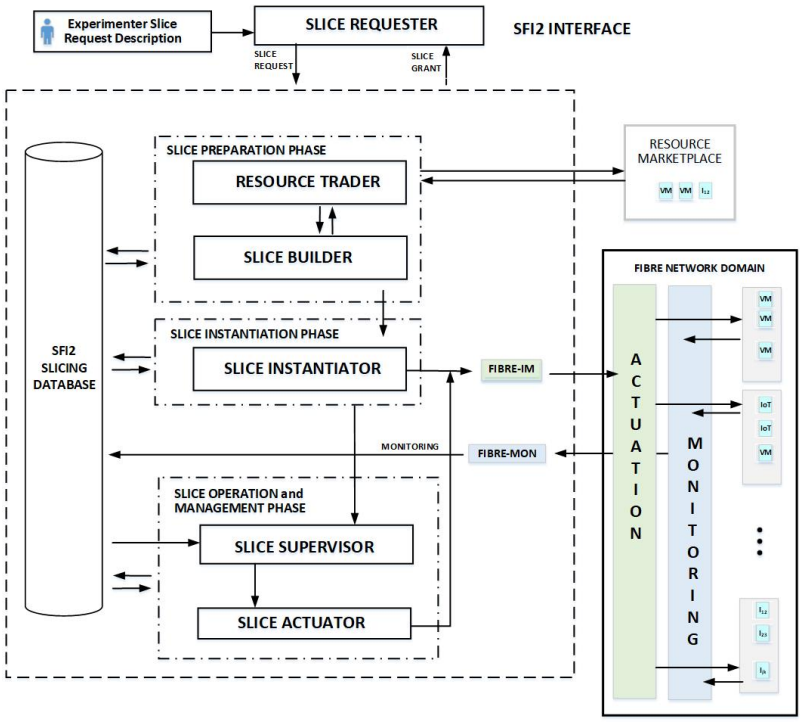

Fig. 2. The SFI2 (Slicing Future Internet Infrastructures) Network Slicing Reference Architecture.

Finally, as the name suggests, the slice supervisor manages the slice operation, allowing slice reconfiguration resulting from user traffic changes, performance parameters tunning, SLA (Service Level Agreement) adjustments, or from the need to reconfigure slices aiming the optimized use of resources by the slice provider (SFI2).

2.2 The Architectural Features Concept for Network Slicing

The architectural feature concept for network slicing can be understood as follows:

- A set of objectives and characteristics defined for the deployed virtual slice and the network slicing architecture as a whole in a slicing process.

To illustrate the concept, it follows a set of non-exhaustive architectural features considered for the scope of the discussion in this paper:

- Resource selection in the preparation phase;
- Security capabilities for the slice or the architecture;
- Optimization of resources for the slices; and
- Optimization of resources for the network slicing provider.

As far as these architectural features are concerned, there is an inherent need for the orchestration among components and other elements in the architecture to achieve the required characteristics or to optimize resources. For example, orchestration for optimizing resources for a slice may consider service profile prediction mechanisms in the context of the slice itself, service profile prediction for the set of slices hosted in a provider, and the overall provider resource

distribution among slices currently in use. When considering different types of attacks, an architecture security service should evaluate and consider a number of different architecture components since many of them are vulnerable to a single type of attack [12]. In summary, architectural features setup requires a suitable orchestration mechanism in the network slicing architectures.

2.3 Architectural Features Orchestration in Network Slicing - Current Scenario

Current alternatives for network-slicing architectures include architectures defined by standardization bodies and research projects. Table 1 illustrates how some of the most relevant architectures inherently consider or not in their deployments a set of architectural features, including:

- The capability to make multiple choices among available resources and optimize them at the preparation phase using the marketplace functionality;
- The capability to make multiple choices among available resources and optimize them at the preparation phase with multiple domains;
- The capability to orchestrate physical and logical resources towards optimization;
- The capability to orchestrate architectural components towards improved security capabilities at slice and architecture levels;
- The capability to orchestrate the composition of slice resources toward optimization; and
- The capability to orchestrate provider resources towards either architecture or provider resources optimization.

Table 1. **Summary of architectural features orchestration in current network slicing architectures (NE - Not explicitly defined)**.

ARCHITECTURAL FEATURE	SFI2	3GPP	ITU-T	ETSI	IETF	NECOS	NASOR
Marketplace	●	○	○	○	○	●	○
Multi-domain	●	●	●	●	●	●	●
Physical/virtual Resource Orchestration	●	NE	NE	NE	NE	●	●
Security Orchestration	●	NE	◑	NE	○	○	○
Slice Resource Orchestration	●	●	●	●	●	●	●
Provider Resource Orchestration	●	○	○	○	○	○	●

The network slicing surveys in [22] [2] [17] [6] and the network slicing architectures presented in [1], [5], [7], [9], [10], [11], and [13] further detail the existing architectural features orchestration for NS. In Table 1, security orchestration refers to the ability to orchestrate security functionalities not only for slices, but also for the architecture that provides these slices. Slice orchestration refers to the ability to orchestrate available resources in slice deployment, whereas provider orchestration refers to the capability to orchestrate resources among deployed slices in an NS architecture. We use the symbols ●, ○, and ◑to represent compliance with the feature, noncompliance, and partial compliance, respectively.

3 ARCHITECTURAL FEATURES ORCHESTRATION FOR NETWORK SLICING ARCHITECTURES - POSITION POINTS AND RECOMMENDATIONS

Network slicing architectures and derived systems must be dynamic and efficient regarding resource orchestration and allocation. To achieve such characteristics, architectural features like the ones indicated in Table 1 must be in place, orchestrating the available resources.

In this regard, we propose the following recommendations for network-slicing architectures that will allow optimization and support architectural features deployment for slice providers and users (Figure 3):

- Machine learning agents as an embedded and intrinsic functional component;
- Federated learning as a network-slicing architectural intelligence capability; and
- Data-driven methods to manage network slicing services.

Having machine learning agents as an embedded and intrinsic functional component means, in other words, including ML-native functionalities in the architecture. This is achieved, for instance, in the SFI2 reference architecture by having the basic phases (preparation, trading, building, operation, supervision, and security) of the network slicing process assisted by ML agents (Figure 3).

However, embedding machine learning agents in a network-slicing architecture solves part of the objective to include a problem-solving intelligence we want to address. In effect, the different slicing phases in the network-slicing process must interact through the specific orchestration solution adopted in the architecture or corresponding deployment. In technical terms, this means that different ML agents in distinct architecture components have models that should be integrated in the best possible way. To attain this objective, we suggest that federated learning should be used to weigh among agents and arrive at a kind of weighted model for the specific orchestration in place.

For example, providing clean and energy-efficient slices across multi-domain resource providers requires the orchestration between distinct energy-efficient algorithms and approaches by providers that should be weighted with respect to the type of clean or not-clean energy they use. In summary, information and knowledge are distributed and, as such, should be used considering this characteristics.

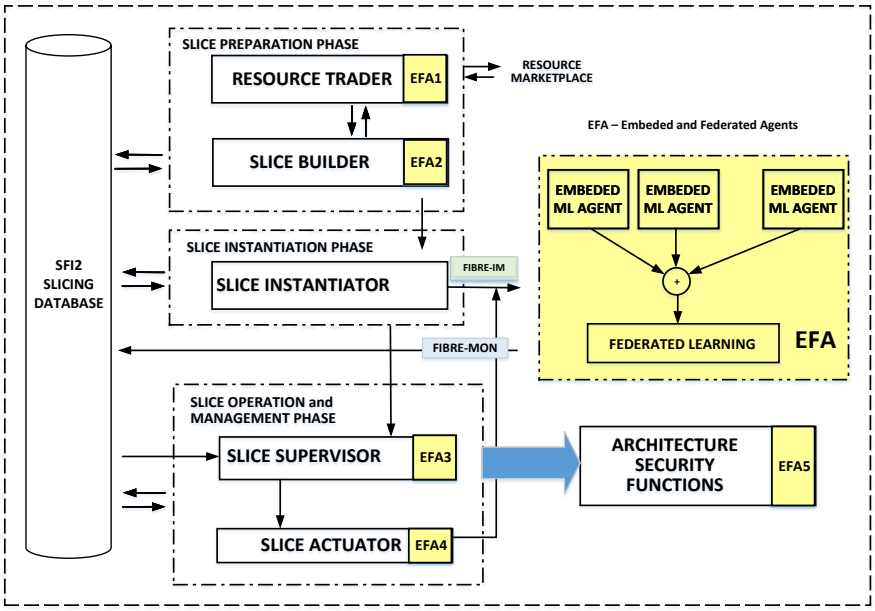

Fig. 3. ML-Native Agents and Federated Learning Recommendations for Architectural Features Orchestration.

4 DATA-DRIVEN METHODS RECOMMENDATION FOR NETWORK SLICING

Data management is a critical element in various areas, including the network architecture. Network slicing generates a large volume of data during different orchestration phases, and the service runtime on top of tailored resources is the primary data source. Network providers can utilize data-driven methods to manage their services and improve security and intelligent network-slicing services. With a vast amount of data available, modern network architectures can offer numerous insights and customized services. To achieve this, we propose a data pipeline, as shown in Figure 4, that includes an Embedded Monitoring Agent (EMA) to collect data and facilitate data ingestion. Services such as Kafka handle data streams for batch and data processing, whereas later services such as Spark distribute streamlined data processing, enabling real-time analytics and seamless scalability. This approach shifts data management practices and fosters agile decision-making in network markets. In addition, some processing can feed EFAs or provide a data science playground for researchers through the Jupyter environment. Ultimately, the network-slicing user or management can gain insights and effectively monitor network slices.

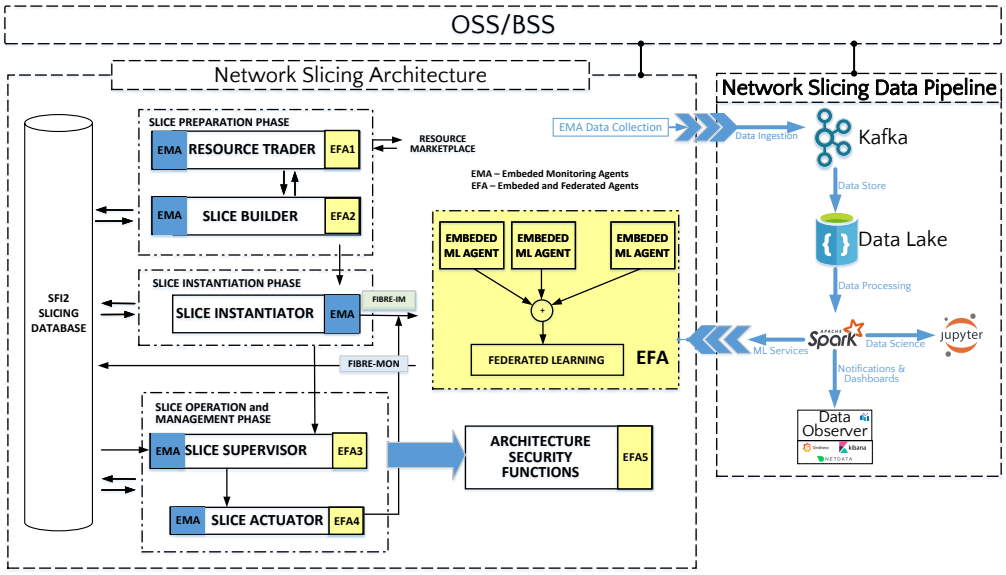

Fig. 4. Architectural Data-Driven Recommendation for Network Slicing.

Granular insights facilitate a novel approach to managing network resources and sharing legacy facilities via the Operations Support System and Business Support System (OSS/BSS). In addition, the data generated by both the applications running through network slicing and the operation of the network architecture can feed into cognitive methods to make communication seamless and intelligent. Many challenges concerning security, data management, and heterogeneous data sources must be considered for next-generation network-slicing methods.

5 ML-NATIVE SECURITY - AN ARCHITECTURAL FEATURE ORCHESTRATION CASE

The SFI2 slicing architecture is an edge-cutting approach that deploys intelligent, energy-efficient network slices while guaranteeing security at both the operational and service levels. Operational security is related to the safety of the

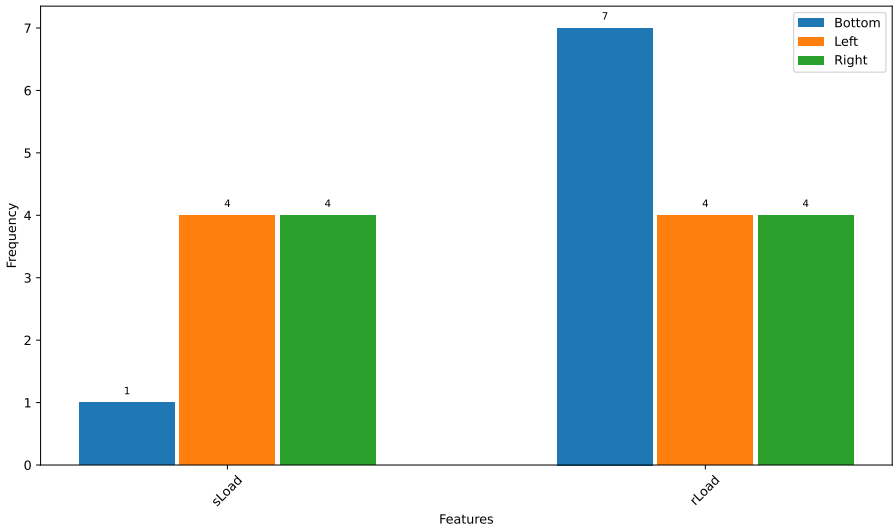

Fig. 5. Frequency of the most important features in whole Westermo dataset.

architectural building block, whereas service refers to the security of the slicing service. Here, we highlight some of the key points of the architecture, particularly the mechanisms that enable the deployment of intelligent and secure network slices [11]. The rationale behind our architecture inaugurates the native distributed machine-learning mechanism embedded into architecture building blocks.

We devised a Machine Learning Agent (ML-Agent) mechanism, which has a dual responsibility: to perceive and act in the environment, the partner iterates over the data collected from the archive to train ML models in a distributed manner and report model weights to the main model in SFI2 AI Management. The mechanism of action in the intelligent network slicing process refers to the interaction between the ML-agent and network slicing functional blocks (Slice Builder, Supervisor, and Actuator in Fig. 3). In this way, for each life cycle of the network slice, the process can rely on machine learning models to make decisions such as resource allocation, provisioning, and Quality of Experience or Service (QoE/QoS) forecasting.

We examined three datasets simultaneously, and applied feature engineering to determine the most important features for data clustering. We applied the elbow method [3] to determine a suitable number for clustering our data using the k-means algorithm, which revealed eight (8) clusters. In the feature engineering scheme shown in Figure 5, we noticed that *rLoad* and *sLoad* were the most important features that best clustered most of the data among the three datasets. The comparison illustrates the varying frequencies of essential features among datasets, implying potential distinctions and similarities within the data.

Knowing the most relevant features identified by the k-means clustering algorithm provides valuable insights for further analysis and model refinement. Regarding slicing architectures, feature orchestration is important once service-level agreements and threat systems can operate earlier, avoiding slices and architecture operation outages. Although these features may not exhibit strong linear correlations, their significance in cluster formation implies they carry essential information for distinguishing distinct groups within the dataset. For further investigations, we believe

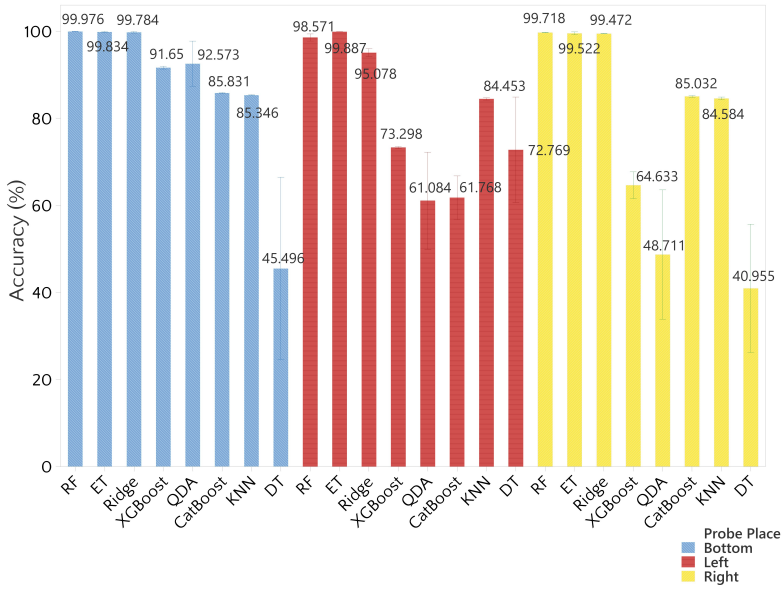

Fig. 6. Localized Test Accuracy (%) achieved by three different ML-Agents in the SFI2 Slicing Architecture.

integrating data pipelining into network-slicing architectures will shift slicing management and orchestration in future network-slicing architectures.

Principal Component Analysis (PCA) was used to identify the most relevant features for each cluster. PCA is a dimensionality reduction technique that transforms data into a new coordinate space, where the axes are called principal components. The first principal component explains most of the data variance and so on. PCA allows us to obtain the coefficients of the principal components that indicate the weight of each original feature in forming the new axes. The feature with the highest absolute coefficient for each principal component is the most relevant for that axis.

Second, we showcase the architecture feature to handle training on different blocks of the SFI2 architecture using a Westermo dataset in our evaluation [19]. This dataset refers to 90 min of packet industrial network slicing, including harmless SSH, bad SSH, misconfigured IP addresses, duplicated IP addresses, port scans, and man-in-the-middle attacks. The context of the dataset refers to an industrial network, and three collections of PCAPs (Packet Capture) probes were distributed along the topology. Based on this dataset, we imported it into the architecture to validate the distributed training and prediction mechanism. We simultaneously considered and trained three datasets, implying a Non-Informally, Identically Distributed (non-IID) configuration. We employed traditional machine learning algorithms to assess the prediction suitability of the ML-Agent among the architectural block operations.

We idealized an experiment considering machine learning algorithms to choose the best one that fits the network data empirically. Hence, our considered network topology has different equipment, a network manufacturing traffic simulator, and three packet probes at different network locations: bottom, left, and right. We empirically evaluated the following algorithms: extra threats classifier (ET), Random Forest Classifier (RF), Ridge, Quadratic Discriminant Analysis (QDA), Extreme Gradient Boost (XGBoost), CatBoost Classifier, K-Neighbors Classifier (KNN), and Decision Tree (DT) [16] [4] [23] [18]. Each ML agent is placed over the network to evaluate the algorithms.

In summary, our experiments considered distributed training over three different ML-Agents for prediction in different architectural layers. We measured the convergence accuracy of the server model using local training and testing. Hence, we summarized our results concerning accuracy and loss over epochs according to Figure 6, where, despite the challenging training scenario with a non-IID dataset, the SFI2 AI Management block handles a model with the prediction of harmless packets with a lower error and 95% confidence. Using locally trained models, the ML-Agent achieved higher accuracies of approximately 99% for many algorithms. We conducted our experiments by considering the stratified KFold with ten (10) folds for each algorithm. This experiment validated the safety features of our architecture while exploring the ML-native slicing architecture. Unlike the QDA and XGBoost algorithms, which performed well in some places of topology, but not at all.

6 FINAL CONSIDERATIONS

Dynamic and optimized orchestration of resources is the main driver of network slicing, which allows its adequacy and, at least in part, justifies its trend in areas such as the next-generation mobile network (5G/6G), IoV, and IIoT, where user requirements are highly stringent and heterogeneous services are required. This study proposes, highlights, and demonstrates that network-slicing architectures should incorporate ML-native agents in their structure, adopt a distributed learning strategy for acquiring knowledge, and incorporate a data-driven method to manage network-slicing services.

A *security service* is an architectural feature demonstrated in the SFI2 architecture, in which embedded security agents use federated learning to acquire cooperative knowledge using a data-driven approach to provide intrusion detection. The architectural deployment for the SFI2 architecture can be replicated in other network slicing architectures by adapting the proposed approach and method demonstrated for the slicing phases and components in the target architecture. For future directions and research agenda, we believe that the coexistence of security, monitoring, and intelligent agents embedded in architectural services as daemons will be crucial for future generations of network slicing architectures and services. In addition, we guess that the hybrid and collaborative use of supervised and unsupervised learning paradigms is essential for discovering knowledge and generating decision-making insights for near-real-time network orchestrators and managers.

ACKNOWLEDGMENTS

- The authors thank CNPq and FAPESP MCTIC/CGI cooperation agreement under the thematic research project 2018/23097-3 - SFI2 - Slicing Future Internet Infrastructures.
- The authors thank the ANIMA INSTITUTE for scholarship support.

REFERENCES

[1] 3GPP. 2019. *3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Management and Orchestration; Concepts, Use Cases and Requirements.* Technical Specification 3GPP TS 28.530 V15.0. 3GPP. 30 pages.

[2] Alcardo Alex Barakabitze, Arslan Ahmad, Rashid Mijumbi, and Andrew Hines. 2020. 5G Network Slicing Using SDN and NFV: A Survey of Taxonomy, Architectures and Future Challenges. *Computer Networks* 167 (Feb. 2020), 106984. https://doi.org/10.1016/j.comnet.2019.106984

[3] Purnima Bholowalia and Arvind Kumar. 2014. EBK-Means: A Clustering Technique based on Elbow Method and K-Means in WSN. *International Journal of Computer Applications* 105, 9 (Nov. 2014), 17–24. https://www.ijcaonline.org/archives/volume105/number9/18405-9674 Publisher: Foundation of Computer Science (FCS).

[4] Archana Chaudhary, Savita Kolhe, and Raj Kamal. 2016. An Improved Random Forest Classifier for Multi-Class Classification. *Information Processing in Agriculture* 3, 4 (Dec. 2016), 215–222. https://doi.org/10.1016/j.inpa.2016.08.002

[5] S Clayman, A Neto, F Verdi, S Correa, S Sampaio, I Sakelariou, L Mamatas, R Pasquini, K Cardoso, F Tusa, C Rothenberg, and J Serrat. 2021. The NECOS Approach to End-to-End Cloud-Network Slicing as a Service. *IEEE Communications Magazine* 59, 3 (March 2021).

[6] Adnei Donatti, Sand L. Correa, Joberto S. B. Martins, Antonio Abelem, Cristiano B. Both, Flavio Silva, José A. Suruagy, Rafael Pasquini, Rodrigo Moreira, Kleber V. Cardoso, and Tereza C. Carvalho. 2023. Survey on Machine Learning-Enabled Network Slicing: Covering the Entire Life Cycle. *IEEE Transactions on Network and Service Management* 21, 3 (2023), 1–18. https://doi.org/10.1109/TNSM.2023.3287651 Conference Name: IEEE Transactions on Network and Service Management.

[7] ETSI. 2015. *Mobile Edge Computing A key technology towards 5G.* Technical Report WP No. 11. European Telecommunications Standards Institute.

[8] Een-Kee Hong, Inkyu Lee, Byonghyo Shim, Young-Chai Ko, Sang-Hyo Kim, Sangheon Pack, Kyunghan Lee, Sunwoo Kim, Jae-Hyun Kim, Yoan Shin, Younghan Kim, and Haejoon Jung. 2022. 6G R&D Vision: Requirements and Candidate Technologies. *Journal of Communications and Networks* 24, 2 (April 2022), 232–245. Conference Name: Journal of Communications and Networks.

[9] IETF. 2021. *Framework for IETF Network Slices.* RFC- Request for Comments draft-ietf-teas-ietf-network-slice-framework-00. Internet Engineering Task Force. 1–18 pages.

[10] Telecommunication Standardization ITU-T. 2012. *Framework of Network Virtualization for Future Networks.* Technical Report ITU-T Y.3011. ITU-T. 1–28 pages.

[11] Joberto S. B. Martins, Tereza C. Carvalho, Rodrigo Moreira, Cristiano Bonato Both, Adnei Donatti, João H. Corrêa, José A. Suruagy, Sand L. Corrêa, Antonio J. G. Abelem, Moisés R. N. Ribeiro, José-marcos S. Nogueira, Luiz C. S. Magalhães, Juliano Wickboldt, Tiago C. Ferreto, Ricardo Mello, Rafael Pasquini, Marcos Schwarz, Leobino N. Sampaio, Daniel F. Macedo, José F. De Rezende, Kleber V. Cardoso, and Flávio De Oliveira Silva. 2023. Enhancing Network Slicing Architectures With Machine Learning, Security, Sustainability and Experimental Networks Integration. *IEEE Access* 11 (2023), 69144–69163. https://doi.org/10.1109/ACCESS.2023.3292788 Conference Name: IEEE Access.

[12] Rodrigo Moreira, Joberto Martins, and Flávio de Oliveira Silva. 2023. On Enhancing Network Slicing Life-Cycle through an AI-native Orchestration Architecture. https://doi.org/10.36227/techrxiv.22293877.v1

[13] Rodrigo Moreira, Pedro Frosi Rosa, Rui Luis Andrade Aguiar, and Flávio de Oliveira Silva. 2021. NASOR: A network slicing approach for multiple Autonomous Systems. *Computer Communications* 179 (2021), 131–144. https://doi.org/10.1016/j.comcom.2021.07.028

[14] Ali Nauman, Tu N. Nguyen, Yazdan A. Qadri, Zulqar Nain, Korhan Cengiz, and Sung Won Kim. 2022. Artificial Intelligence in Beyond 5G and 6G Reliable Communications. *IEEE Internet of Things Magazine* 5, 1 (March 2022), 73–78. https://doi.org/10.1109/IOTM.001.2100140 Conference Name: IEEE Internet of Things Magazine.

[15] Hnin Pann Phyu, Diala Naboulsi, and Razvan Stanica. 2023. Machine Learning in Network Slicing—A Survey. *IEEE Access* 11 (2023), 39123–39153. https://doi.org/10.1109/ACCESS.2023.3267985 Conference Name: IEEE Access.

[16] Iqbal H. Sarker. 2021. Machine Learning: Algorithms, Real-World Applications and Research Directions. *SN Computer Science* 2, 3 (March 2021), 160. https://doi.org/10.1007/s42979-021-00592-x

[17] Xuemin Shen, Jie Gao, Wen Wu, Kangjia Lyu, Mushu Li, Weihua Zhuang, Xu Li, and Jaya Rao. 2020. AI-Assisted Network-Slicing Based Next-Generation Wireless Networks. *IEEE Open Journal of Vehicular Technology* 1 (2020), 45–66. https://doi.org/10.1109/OJVT.2020.2965100 Conference Name: IEEE Open Journal of Vehicular Technology.

[18] Yan-yan SONG and Ying LU. 2015. Decision Tree Methods: Applications for Classification and Prediction. *Shanghai Archives of Psychiatry* 27, 2 (April 2015), 130–135. https://doi.org/10.11919/j.issn.1002-0829.215044

[19] Per Erik Strandberg, David Söderman, Alireza Dehlaghi-Ghadim, Miguel Leon, Tijana Markovic, Sasikumar Punnekkat, Mahshid Helali Moghadam, and David Buffoni. 2023. The Westermo network traffic data set. *Data in Brief* 50 (2023), 109512. https://doi.org/10.1016/j.dib.2023.109512

[20] Prashant Subedi, Abeer Alsadoon, P. W. C. Prasad, Sabih Rehman, Nabil Giweli, Muhammad Imran, and Samrah Arif. 2021. Network Slicing: A Next Generation 5G Perspective. *EURASIP Journal on Wireless Communications and Networking* 2021, 1 (April 2021), 102. https://doi.org/10.1186/s13638-021-01983-7

[21] Abdul Waheed, Munam Ali Shah, Syed Muhammad Mohsin, Abid Khan, Carsten Maple, Sheraz Aslam, and Shahab Shamshirband. 2022. A Comprehensive Review of Computing Paradigms, Enabling Computation Offloading and Task Execution in Vehicular Networks. *IEEE Access* 10 (2022), 3580–3600. Conference Name: IEEE Access.

[22] Yulei Wu, Hong-Ning Dai, Haozhe Wang, Zehui Xiong, and Song Guo. 2022. A Survey of Intelligent Network Slicing Management for Industrial IoT: Integrated Approaches for Smart Transportation, Smart Energy, and Smart Factory. *IEEE Communications Surveys & Tutorials* 24, 2 (2022), 1175–1211. https://doi.org/10.1109/COMST.2022.3158270 Conference Name: IEEE Communications Surveys & Tutorials.

[23] Zhongheng Zhang. 2016. Introduction to Machine Learning: K-Nearest Neighbors. *Annals of Translational Medicine* 4, 11 (June 2016), 218. https://doi.org/10.21037/atm.2016.03.37

A RESEARCH METHODS

A.1 Part One

The research method used in this paper is exploratory and quantitative. An exploratory method is achieved by analyzing, discussing, and recommending intelligent architectural feature orchestration for network slicing architectures. The quantitative method is achieved by simulating a case for embedding security in the SFI2 architecture.

A.2 Part Two

The security architectural feature orchestration case simulates three (3) ML agents deployed in distinct SFI2 architecture functional blocks (phases) with perceive and act capabilities. The agents are trained using the extra threats classifier (ET), Random Forest Classifier (RF), Ridge, Quadratic Discriminant Analysis (QDA), Extreme Gradient Boost (XGBoost), CatBoost Classifier, K-Neighbors Classifier (KNN), and Decision Tree (DT) ML-classification methods with the Westermo dataset over relevant features detected. The federated learning average approach is used to measure convergence accuracy for the distributed training.

B ONLINE RESOURCES

The security architectural feature case experimental dataset and code are available on GitHub: https://github.com/romoreira

Received 08 December 2023; revised xx January 2024; accepted xx February 2024