A Prototype of O-RAN-Based End-to-End Network Slice Management with Closed-Loop RAN Control

Tse-Han Wang*†, Cheng-Yu Huang*, Li-Hsing Yen*, and Chien-Chao Tseng*
*Department of Computer Science, College of Computer Science,
National Yang Ming Chiao Tung University, Hsinchu, Taiwan.
Email: {wangth, cyhuang0415, lhyen, cctseng}@cs.nycu.edu.tw
†Network Management Laboratory, Chunghwa Telecom Laboratories, Taoyuan, Taiwan.

***Abstract*—This paper presents a prototype implementation of an O-RAN-based end-to-end network slicing management and orchestration framework with closed-loop control. The proposed framework integrates ETSI NFV-MANO and the O-RAN Service Management and Orchestration (SMO) architecture to coordinate slice lifecycle management across access, transport, and core network domains using Kubernetes-based orchestration. OpenAirInterface (OAI) and free5GC are adopted as open-source radio access network (RAN) and core network platforms, respectively. Closed-loop RAN slice management is implemented through a Non-Real-Time RAN Intelligent Controller (Non-RT RIC) rApp using O1-based monitoring and O2-based horizontal scaling orchestration. Experimental results demonstrate practical E2E slice deployment with limited orchestration overhead and highlight the operational complexity of RAN lifecycle management and multi-domain coordination in cloud-native O-RAN environments.**

***Index Terms*—5G, O-RAN, Network Slicing, Cloud-Native Orchestration, Service Management and Orchestration**

I. Introduction

Fifth-generation (5G) mobile networks support heterogeneous services with diverse performance requirements. Network slicing enables multiple logical networks with differentiated service characteristics to share common physical infrastructure [1], [2]. The O-RAN architecture extends the disaggregated Next-Generation Radio Access Network (NG-RAN) with standardized management and control interfaces, enabling cloud-native deployment and closed-loop automation through the Service Management and Orchestration (SMO) framework and RAN Intelligent Controllers (RICs) [3], [4].

Coordinating end-to-end (E2E) network slices across RAN, transport, and core domains remains operationally challenging. Although prior studies have explored cloud-native slicing and multi-domain orchestration, publicly documented prototypes jointly integrating O-RAN SMO-based RAN management, ETSI Network Functions Virtualization Management and Orchestration (NFV-MANO), and closed-loop RAN control within a unified E2E slicing environment remain limited.

This paper presents a prototype implementation of an O-RAN-based E2E network slicing management and orchestration framework. Building upon our previous E2E slicing system [5], the proposed framework integrates ETSI NFV-MANO with O-RAN SMO to coordinate lifecycle management across RAN, transport, and core domains. O-RAN network functions, including O-RAN Central Unit Control Plane (O-CU-CP), O-RAN Central Unit User Plane (O-CU-UP), and O-RAN Distributed Unit (O-DU), are implemented using the OpenAirInterface (OAI) RAN platform [6], while Kubernetes Operators automate network function and subnet slice lifecycle management.

Closed-loop RAN slice control is implemented using a Non-Real-Time RIC (Non-RT RIC) rApp. Runtime metrics and fault events collected through the O1/VNF Event Stream (VES) interface are evaluated using predefined policies, and horizontal scaling is triggered through the O2 interface. Runtime adaptation is limited to the RAN domain, while the transport domain provides VXLAN connectivity and Linux Traffic Control (TC)-based bandwidth enforcement.

The main contributions are summarized as follows:

- **Standards-aligned E2E slice orchestration:** An E2E slicing prototype integrating ETSI NFV-MANO and O-RAN SMO across RAN, transport, and core domains, including VXLAN-based connectivity and TC-based bandwidth enforcement.
- **Cloud-native O-RAN lifecycle management:** A Kubernetes Operator-based framework supporting lifecycle management of O-RAN network functions and RAN subnet slices.
- **Policy-driven closed-loop RAN control:** A Non-RT RIC rApp-based control workflow using O1/VES monitoring and O2-driven horizontal scaling.

II. Related Work

Early studies on E2E network slicing primarily focused on orchestration architectures, slice customization, and cross-domain coordination across the RAN, transport, and core network domains [7], [8], [9], [10], [11], [12].

Subsequent research adopted cloud-native technologies, including containerized network functions, Kubernetes orchestration, and NFV-based lifecycle management. Our previous work proposed a cloud-native E2E slicing framework integrating transport and core network management and later extended it with MEC support [5], [13]. Building upon these studies, this work further incorporates O-RAN SMO-based RAN lifecycle management and policy-driven closed-loop control into the existing E2E slicing framework.

In parallel, O-RAN research has investigated RAN slicing, RIC-based automation, and SMO-based management. Existing studies have primarily focused on RAN-specific management and optimization using open-source platforms such as OAI RAN. In contrast, this work focuses on standards-aligned integration of ETSI NFV-MANO and the O-RAN SMO framework to enable coordinated E2E slice management and closed-loop RAN control.

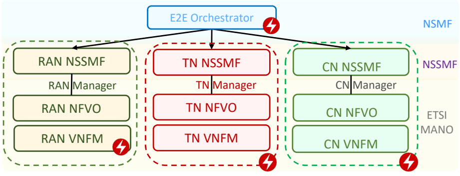

Fig. 1: Mapping the proposed E2E slicing framework to the 3GPP management framework and ETSI NFV-MANO architecture

III. E2E Slice Orchestration and Closed-Loop RAN Management

This section presents the proposed E2E network slicing management and orchestration framework together with the closed-loop control mechanism for RAN slice management. The framework integrates the 3GPP management architecture, ETSI NFV-MANO, and the O-RAN SMO framework to coordinate slice lifecycle management across the RAN, transport, and core network domains.

A. System Architecture Overview

Fig. 1 shows the mapping of the proposed E2E slicing framework to the 3GPP management framework and ETSI NFV-MANO while incorporating O-RAN management components for RAN slice management. The E2E Orchestrator serves as the Network Slice Management Function (NSMF) and coordinates slice lifecycle management across the RAN, transport, and core network domains. The Core Network (CN) Manager and Transport Network (TN) Manager manage subnet slices within their respective domains, while the RAN domain follows the O-RAN architecture, where the RAN NSSMF, Network Function Orchestrator (NFO), and Deployment Management Services (DMS) are implemented as independent components.

All management components are implemented as Kubernetes Operators [14]. Upon receiving an E2E slice request, the E2E Orchestrator coordinates subnet slice provisioning across the CN, TN, and RAN domains through REST-based orchestration interfaces. The TN Manager further coordinates TN agents deployed on the central and edge nodes to establish VXLAN tunnels and configure slice-specific bandwidth policies using Linux TC.

Compared with the transport and core domains, cloud-native RAN slicing introduces additional orchestration complexity because of strict timing constraints, stateful protocol dependencies, and limited cloud-native support in the adopted OAI implementation.

B. O-RAN-based RAN Slice Management

The access network is implemented using OAI RAN v2.0.0. Since the adopted implementation does not natively support CU control and user plane separation, this work extends OAI to support the E1 interface between the O-CU-CP and O-CU-UP. However, the current implementation does not support one-to-many CU-CP to CU-UP state management, and both O-CU-CP and O-CU-UP accept only a single O-DU connection. Consequently, each RAN slice consists of one O-CU-CP, one O-CU-UP, and one O-DU.

Fig. 2 presents the RAN slice management architecture. The RAN NSSMF manages subnet slice lifecycle requests

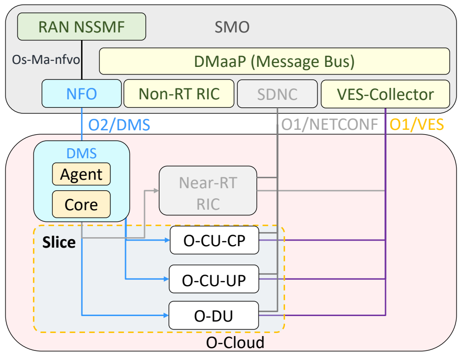

Fig. 2: O-RAN-based RAN slice management architecture

and coordinates with the NFO, which performs resource orchestration and network function instantiation through the DMS. The DMS is implemented as a Kubernetes-based VNFM responsible for lifecycle management of RAN network function instances through Kubernetes custom resources and O-RAN Operators.

1) Descriptor and Template Model: RAN subnet slices are described using ETSI- and 3GPP-aligned descriptors. Virtual Network Function Descriptors (VNFDs) specify the resource requirements of individual RAN network functions, Network Service Descriptors (NSDs) define subnet slice composition by referencing the required VNFDs, and Network Resource Models (NRMs) describe slice-level parameters including Public Land Mobile Network (PLMN) ID, Single Network Slice Selection Assistance Information (S-NSSAI), and service requirements. VNFDs and NSDs are represented using TOSCA templates, whereas NRMs are represented in JSON and are used during subnet slice instantiation and lifecycle management.

2) RAN Slice Provisioning Workflow: Fig. 3 illustrates the operational workflow of RAN slice provisioning, which consists of three phases: preparation, creation, and activation.

During the preparation phase, VNFDs, NSDs, and NRMs are uploaded to the RAN NSSMF and stored for subsequent provisioning requests. During the creation phase, the RAN NSSMF determines whether existing subnet slices satisfy the requested service requirements. If additional resources are required, it requests the NFO to instantiate a new subnet slice. The NFO subsequently allocates resources and coordinates with the DMS to create the required RAN network function instances while maintaining the operational state of the subnet slice. During the activation phase, the O-RAN Operator within the DMS deploys the requested RAN network functions through Kubernetes-based lifecycle management.

The final network configuration stage defined in the workflow is currently limited by the adopted OAI implementation, which does not fully support NETCONF-based RAN configuration management.

C. Closed-loop RAN Slice Control

The proposed framework implements policy-driven closed-loop RAN slice control through the O-RAN SMO framework, as illustrated in Fig. 4. After a RAN slice is instantiated, the NFO registers the corresponding slice information with the Non-RT RIC rApp to enable runtime monitoring and policy-based resource management.

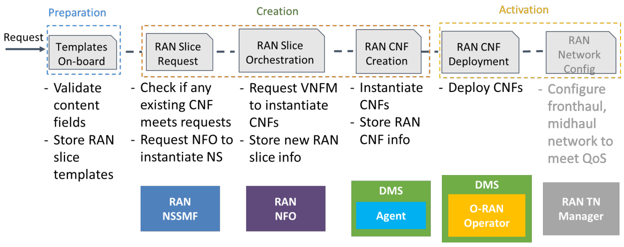

Fig. 3: Operational workflow of RAN slice allocation

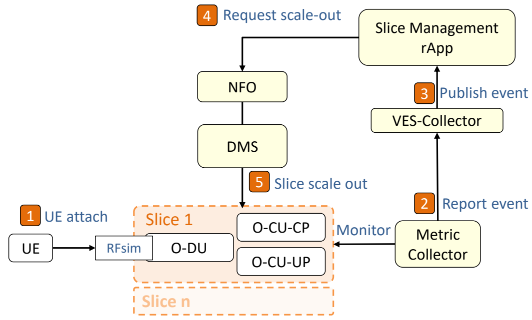

Fig. 4: Closed-loop control flow for RAN slice management

A Metric Collector deployed alongside each RAN slice periodically collects runtime metrics and fault events from RAN network functions and reports them to the VES Collector through the O1/VES interface. The collected events are distributed through DMaaP and analyzed by the slice management rApp running on the Non-RT RIC.

Based on predefined operational policies, the rApp determines whether additional resources are required. When overload conditions are detected, the rApp issues O2-based scaling requests toward the DMS orchestration layer, which subsequently coordinates Kubernetes-based lifecycle management procedures to instantiate additional RAN network function instances.

Transport network slicing is implemented using VXLAN tunnels and TC-based bandwidth policies coordinated by the TN Manager and TN agents, while the core network is implemented using free5GC with a shared/dedicated network function design [5]. Due to limitations of the adopted OAI implementation, the current prototype focuses on O2-based RAN lifecycle management and O1/VES monitoring.

IV. Evaluation

This section evaluates the proposed prototype from an operational perspective. The evaluation focuses on E2E slice deployment overhead, closed-loop RAN slice control, and the operational characteristics of the implemented framework.

A. Experimental Setup

Experiments were conducted using three virtual machines (VMs), each deployed on a separate physical server. The environment consists of three single-node Kubernetes clusters representing the management, central, and edge domains of the proposed E2E slicing framework.

The management node hosts the SMO framework, the E2E Orchestrator, and the TN Manager. The central node deploys the core network manager and free5GC, while the edge node hosts the RAN VNFM, OAI RAN, Metric Collector, and TN agent. Runtime monitoring data are reported to the management node through the O1/VES interface.

Table I summarizes the hardware and software configurations, and Fig. 5 illustrates the experimental environment.

TABLE I: Experimental Environment Configuration

Node	Software	VM	Physical Server
Mgmt.	Ubuntu 20.04 O-RAN SMO H-Release Kubernetes v1.21.6	10 vCPU 32 GB RAM	Intel Xeon Gold 5218 32 cores @2.30 GHz 192 GB RAM
Central	Ubuntu 18.04 free5GC v3.2.1 Kubernetes v1.21.6	10 vCPU 16 GB RAM	Intel Xeon E5-2630 v4 40 cores @2.20 GHz 128 GB RAM
Edge	Ubuntu 18.04 Kubernetes v1.21.6 OAI RAN v2.0.0	20 vCPU 32 GB RAM	Intel Xeon E5-2630 v4 40 cores @2.20 GHz 128 GB RAM

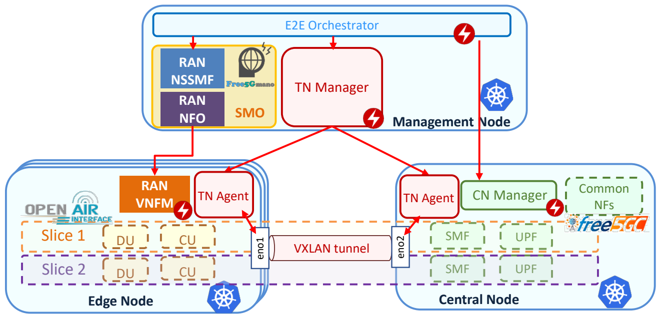

Fig. 5: Experimental environment

B. Deployment Overhead

Table II summarizes the deployment time of two E2E network slices. The first slice required approximately 43 seconds because shared and dedicated core network functions together with transport tunnels were instantiated. The second slice reused existing shared resources and required only dedicated network function deployment and TC rule updates, reducing the deployment time to approximately 28 seconds.

The results indicate that E2E slice deployment latency is primarily dominated by RAN slice instantiation and multi-domain coordination.

C. Closed-loop Control Evaluation

To evaluate the closed-loop control mechanism, two E2E network slices were deployed before a UE attached to the first slice. Due to the current OAI RAN implementation, a single DU instance supports only one UE, causing an overload event to be reported through the O1/VES interface.

Upon detecting the overload event, the Non-RT RIC rApp triggered horizontal scaling through the O2 interface. The additional RAN slice instance was deployed in approximately 16 seconds, demonstrating the operational feasibility of policy-driven closed-loop RAN slice management.

TABLE II: Deployment Time of E2E Network Slices

Deployment Object	Details	Time (s)
1st E2E Slice	CN, TN slices	27.0
	RAN slice	15.9
2nd E2E Slice	CN slice	11.8
	RAN slice	16.2

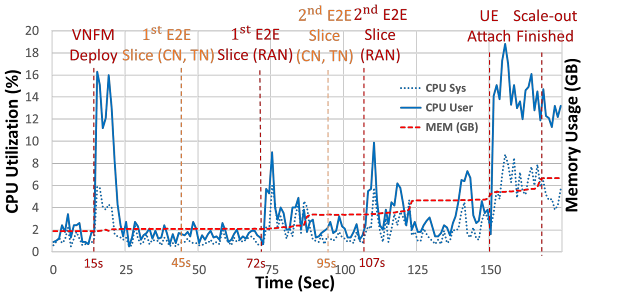

Fig. 6: CPU and memory usage during UE attachment and RAN slice scaling

D. Discussion and Lessons Learned

The experimental results reveal several operational characteristics of the proposed E2E slicing framework.

Limited orchestration overhead: The management and orchestration components introduce relatively low deployment overhead compared with RAN network function deployment. This observation indicates that the orchestration framework itself is not the primary performance bottleneck. Instead, the dominant overhead originates from lifecycle management of cloud-native RAN functions deployed at the edge.

Provisioning latency dominated by RAN lifecycle operations: Multi-domain coordination and RAN lifecycle management account for most of the E2E slice provisioning time. Although transport and core network resources can reuse existing infrastructure, RAN network functions must be instantiated for each new slice in the current implementation. This result suggests that improving cloud-native RAN deployment efficiency is likely to provide greater benefits than further optimizing orchestration logic.

Limitations of runtime closed-loop adaptation: The proposed framework demonstrates the operational feasibility of standards-based closed-loop control using the O-RAN SMO framework. However, the effectiveness of runtime adaptation remains constrained by limitations of the adopted OAI RAN implementation, including restricted CU/DU association flexibility and limited multi-UE support. These implementation constraints currently limit the scalability of elastic RAN slice management.

Standards-based multi-domain orchestration: The prototype validates the integration of ETSI NFV-MANO, the O-RAN SMO framework, and Kubernetes-based orchestration within a unified E2E slicing environment. The implementation experience indicates that lifecycle coordination across heterogeneous management domains remains more challenging than individual domain orchestration, highlighting the importance of standardized interfaces and consistent lifecycle management mechanisms for future cloud-native E2E slicing systems.

V. Conclusions

This paper presented an O-RAN-based prototype for E2E network slice management and orchestration, integrating ETSI NFV-MANO and the O-RAN SMO framework. The proposed system demonstrates standards-aligned E2E slice orchestration and policy-driven closed-loop RAN slice management using open-source cloud-native platforms. Experimental results show that the orchestration framework incurs limited overhead, while RAN lifecycle operations and multi-domain coordination dominate slice provisioning latency.

The implementation also highlights practical limitations in current open-source RAN platforms regarding the flexibility of resource sharing and the controllability of resource granularity. These observations indicate that scalable cloud-native RAN lifecycle management and operational coordination across heterogeneous network domains remain important challenges in practical E2E slicing environments. Future work will focus on extending closed-loop adaptation across transport and core network domains and improving orchestration flexibility for scalable multi-domain E2E slicing environments.

Acknowledgement

This work was supported in part by the Center for Intelligent Team Robotics and Human-Robot Collaboration under the "Top Research Centers in Taiwan Key Fields Program" of the Ministry of Education (MOE), Taiwan, and in part by the National Science and Technology Council, Taiwan, under grants 115-2221-E-A49-004 and 115-2640-E-011-003.

References

[1] "5G; management and orchestration; concepts, use cases and requirements," 3GPP, TS 28.530, V16.2.0, Release 16, Aug. 2020.
[2] "O-RAN study on O-RAN slicing," O-RAN Alliance, TS v02.00, Apr. 2020.
[3] "5G; NG-RAN; architecture description," 3GPP, TS 38.401, V16.3.0, Release 16, Nov. 2020.
[4] "O-RAN operations and maintenance architecture," O-RAN Alliance, TS v07.00, Jul. 2022.
[5] Y.-S. Chiu, L.-H. Yen, T.-H. Wang, and C.-C. Tseng, "A cloud native management and orchestration framework for 5G end-to-end network slicing," in *Proc. 16th IEEE Int'l Conf. on Service-Oriented System Engineering*, CA, USA, Aug. 2022.
[6] "OpenAirInterface," https://openairinterface.org/.
[7] G. Garcia-Aviles, M. Gramaglia, P. Serrano, and A. Banchs, "POSENS: A practical open source solution for end-to-end network slicing," *IEEE Wireless Communications*, vol. 25, no. 5, pp. 30–37, Oct. 2018.
[8] I. Afolabi, T. Taleb, P. A. Frangoudis, M. Bagaa, and A. Ksentini, "Network slicing-based customization of 5G mobile services," *IEEE Network*, vol. 33, no. 5, pp. 134–141, 2019.
[9] Q. Wang, J. Alcaraz-Calero, R. Ricart-Sanchez, M. B. Weiss, A. Gavras, N. Nikaein, X. Vasilakos, B. Giacomo, G. Pietro, M. H. Mark Roddy, P. Walsh, T. Truong, Z. Bozakov, K. Koutsopoulos, P. Neves, C. Patachia-Sultanoiu, M. Iordache, E. Oproiu, I. G. B. Yahia, C. Angelo, C. Zotti, G. Celozzi, D. Morris, R. Figueiredo, D. Lorenz, S. Spadaro, G. Agapiou, A. Aleixo, and C. Lomba, "Enable advanced QoS-aware network slicing in 5G networks for slice-based media use cases," *IEEE Trans. on Broadcasting*, vol. 65, no. 2, pp. 444–453, Jun. 2019.
[10] A. Esmaeily, K. Kralevska, and D. Gligoroski, "A cloud-based SDN/NFV testbed for end-to-end network slicing in 4G/5G," in *Proc. 6th IEEE Conf. on Network Softwarization*, Ghent, Belgium, 2020, pp. 29–35.
[11] X. Li, R. Ni, J. Chen, Y. Lyu, Z. Rong, and R. Du, "End-to-end network slicing in radio access network, transport network and core network domains," *IEEE Access*, vol. 21, no. 8, pp. 29 525–29 537, 2020.
[12] M. Maule, J. Vardakas, and C. Verikoukis, "5G RAN slicing: Dynamic single tenant radio resource orchestration for embb traffic within a multi-slice scenario," *IEEE Communications Magazine*, vol. 59, no. 3, pp. 110–116, 2021.
[13] Y.-C. Chan, L.-H. Yen, T.-H. Wang, and C.-C. Tseng, "MEC Implementation in NFV Architecture Supporting 5G End-to-End Network Slicing," in *2023 2nd International Conference on 6G Networking (6GNet)*, 2023, pp. 1–4.
[14] "Operator framework," https://github.com/operator-framework.