

# 5Guard: Isolation-Aware End-to-End Slicing of 5G Networks

Mehdi Bolourian <sup>✉</sup>, *Graduate Student Member, IEEE*, Noura Limam <sup>✉</sup>, *Member, IEEE*,  
 Mohammad Ali Salahuddin <sup>✉</sup>, *Member, IEEE*, and Raouf Boutaba <sup>✉</sup>, *Fellow, IEEE*

**Abstract**—Network slicing logically partitions the 5G infrastructure to cater to diverse verticals with varying requirements. However, resource sharing exposes the slices to threats and performance degradation, making slice isolation essential. Fully isolating slices is resource-prohibitive, prompting the need for isolation-aware network slicing, where each slice is assigned a tailored isolation level to balance security, usability, and overhead. This paper investigates end-to-end 5G network slicing with resource isolation from the perspective of the infrastructure provider, ensuring compliance with the customers' service-level agreements. We formulate the online 5G isolation-aware network slicing (5G-INS) as a mixed-integer programming problem, modeling realistic slice isolation levels and integrating slice prioritization. To solve 5G-INS, we propose 5Guard, a novel adaptive framework that leverages an ensemble of custom optimization algorithms to achieve the best solution within resource budget and time constraints. Our results show that 5Guard increases profit by up to 15.1% and admission by up to 33.5% in a real-world large-scale network compared to the best-performing individual algorithm. Furthermore, we analyze the trade-offs between isolation levels, their impact on resource utilization, and the effects of slice placement, demonstrating significant advantages over baseline approaches that enforce uniform isolation policies.

**Index Terms**—5G and beyond networks, network slicing, security, mixed-integer programming.

## I. INTRODUCTION

THE fifth-generation (5G) mobile networks, empowered by network slicing (NS), promise to support diverse verticals simultaneously [1], [2], [3] on a shared infrastructure. Due to this sharing, these verticals may need security measures [4] to ensure isolation from vulnerabilities and threats [5], such as distributed denial-of-service (DDoS) or container escape [6]. End-to-end (E2E) isolation is also recognized as a strict demand for sixth-generation (6G) services [7]. Service customers can request isolated slices, or isolation may be mandated by the security requirements of the slice [8].

Among the most effective approaches to offer the highest security is providing physical isolation of 5G resources across

network slices [5], [9]. However, physical isolation for all network slices can impose significant overhead on the infrastructure provider (InP) and reduce resource utilization. Thus, it is crucial to offer different levels of isolation [9], [10] (e.g., physical/complete, logical/semi, or no isolation at all [9]) to balance security and utilization on a need basis [11], [12], [13], [14]. Recent efforts further emphasize the importance of this approach [15], [16], [17]. Consequently, this requires developing new NS schemes that address the above trade-offs in 5G.

A significant challenge in implementing isolation levels within NS lies in ensuring that slices requiring complete isolation are allocated dedicated resources across all segments of the 5G network, which can be hard at scale. One approach to mitigate this challenge is to consolidate resources allocated to other non-fully isolated slices through revisitation, i.e., reusing the same physical entities for realizing these slices, and service migration, which entails transferring services to different physical resources to free up capacity. However, the downtime associated with migration may not be acceptable for all verticals. To address this, slices can be categorized based on priority: high-priority (HP) slices, such as those supporting emergency services, and low-priority (LP) slices, like free public streaming services, and the migration of services enabled in a manner that aligns with the specific requirements and tolerances of each slice.

Another challenge in implementing isolation-aware NS is managing the computational complexity of the mixed-integer resource allocation problem. This complexity can impose scalability and real-world implementation concerns, as exact approaches to tackle this do not often perform well when scaling the network size and the number of slice requests, i.e., the network's load. Additionally, this complexity can introduce substantial delays, which can be especially intolerable for HP slices. Some algorithms may provide worst-case guarantees for solutions; however, a superior guarantee does not ensure better outcomes in all cases. These complexity or optimality trade-offs can limit their practicality in isolation-aware NS systems that support diverse services. In this regard, an ensemble of custom optimization algorithms can be employed, ranging from fast sub-optimal solutions to slower exact (i.e., optimal) ones. These algorithms can run concurrently by leveraging parallel processing to find the best solution within the resource allocation deadline. This deadline can be considered a quality of service (QoS) requirement, which ensures slices receive their service on time, regardless of the number of slice requests (SR) and the size of the 5G infrastructure network.

Received 19 April 2025; revised 15 August 2025; accepted 26 October 2025. Date of publication 31 October 2025; date of current version 6 March 2026. This work was supported in part by funding from the Innovation for Defence Excellence and Security (IDEaS) program from the Department of National Defence (DND). Recommended for acceptance by P. Serrano. (*Corresponding author: Mehdi Bolourian.*)

The authors are with the David R. Cheriton School of Computer Science, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: mbolourian@uwaterloo.ca; noura.limam@uwaterloo.ca; mohammad.salahuddin@uwaterloo.ca; rboutaba@uwaterloo.ca).

Digital Object Identifier 10.1109/TMC.2025.3627066

This research aims to answer the following questions while taking into account the outlined issues and challenges:

- 1) What are the trade-offs of incorporating resource isolation levels into E2E 5G NS?
- 2) How can we formulate these isolation levels for E2E 5G NS?
- 3) What techniques can improve the feasibility and effectiveness of resource isolation in practice?
- 4) How can we achieve an efficient isolation-aware NS that ensures timely service provisioning regardless of network load?

To address these research questions, we first formulate an isolation-aware E2E NS problem, modeling three levels of resource isolation while accounting for trade-offs, overheads, and resource usage patterns based on the recommendations. The formulation employs revisitation and migration techniques to ensure feasibility and scalability for complete isolation. To manage the complexity of the problem, we propose an ensemble of algorithms, each with distinct optimality and complexity traits, to achieve an optimal balance within the resource allocation deadline.

Our main contributions are as follows.

- *Problem Formulation:* We mathematically define the 5G isolation-aware NS (5G-INS) problem. It is the first formulation to incorporate E2E resource isolation levels, usage patterns, and overheads. Our online model ensures proper isolation and QoS for SRs, while enhancing slice acceptance through revisitation, service migration, and controlled QoS violations of LP SRs where permissible.
- *Framework and Algorithm Design:* We propose 5Guard, a novel framework that operates an ensemble of optimization algorithms in parallel at each decision point. These algorithms, designed explicitly for 5G-INS with varying complexity and optimality characteristics, will be selected based on slice priority-imposed deadlines to balance exact and fast solutions.
- *Formal Analysis:* We formally analyze the 5G-INS problem, proving its NP-hardness and evaluating the integrality gap. Additionally, we examine the parallel complexity of 5Guard and demonstrate that the relaxed 5G-INS can be effectively expressed as a scalable linear program with polynomial growth in constraints, thus validating 5Guard's scalability for 5G isolation-aware network slicing.
- *Performance Evaluation and Isolation Analysis:* Through extensive experiments, we first show that 5Guard achieves up to 15.1% higher profit and 33.5% higher slice admission rate compared to the best-performing individual custom-designed 5G-INS algorithms and outperforms the exact algorithm for strict resource allocation deadlines with 100% admission across all isolation levels. Additionally, we analyze how isolation levels and the placement of slices impact resource efficiency and overhead costs compared to baselines with full isolation and no isolation.

The rest of this paper is organized as follows. Section II presents the slice isolation background and explores the related works. Section III describes the modeling of the 5G network, isolation levels, and slice requests. In Section IV, we formulate

the 5G-INS problem and formally analyze its tractability. We propose the 5Guard framework and study its complexity in Section V. In Section VI, we numerically evaluate the 5Guard algorithm, and discuss its limitations. We conclude the paper and instigate future research in Section VII.

## II. BACKGROUND AND RELATED WORKS

In this section, we explore the concept of isolation in 5G NS from different perspectives. We also categorize and review the literature in three related categories: NS, isolation, and isolation-aware NS.

### A. Background

In cybersecurity, isolation aims to minimize an entity's impact on others, which can serve different purposes [18]. From a security perspective, the goal is to prevent an adversarial entity from initiating attacks on other entities. In addition, effective isolation can mitigate privacy issues [19] or eavesdropping. This isolation dimension is typically called *security isolation* [13]. Isolating 5G slices can be achieved in various layers, including physical/virtual resources, protocols, applications, and even programming languages. However, not all of these might be relevant or practical for an InP to implement. For instance, the InP does not control the language used in different applications. The layer in which the isolation is enforced can signify its strength.

From the software-defined networking perspective, we can also isolate slices at management and data planes [13], [18]. At the management plane, isolation is enforced through the concept of multi-tenancy. However, even with isolation at higher layers, the security risk can persist if the underlying resources are shared between slices (e.g., container-escape attacks [6]). Therefore, resource isolation is crucial for achieving inter-slice isolation and can span both the data and management planes.

The degree of resource isolation can be measured by comparing the amount of shared resources between slices [13]. For instance, virtual machines (VM) offer greater isolation than containers due to fewer shared resources (e.g., the operating system (OS)'s kernel). Enforcing isolation or operating in an isolation level can impose additional overheads (e.g., a hypervisor) or resource usage patterns intrinsic to that level and can differ across various 5G resource types, i.e., radio, transport, and mobile-edge computing (MEC)/cloud. For instance, complete (i.e., physical) isolation consumes all the resources of the physical entity, regardless of the resources requested. Studying the impact of resource isolation on 5G NS is one of the main focuses of this work.

### B. Related Works

We explore the related works in three categories as follows.

1) *State-of-the-Art NS:* Traditional 5G and beyond NS has been studied in the literature. A branch of research in this category abstracts NS as virtual network embedding (VNE) or service function chaining (SFC) [23], [24], [25], [26], [27], [28], [29]. The authors in [23] proposed a VNE solution based on graph neural network (GNN) and reinforcement learning

(RL) for dynamic VNE, where both virtual networks (VNs) and physical networks (PNs) are modeled as weighted graphs with changing topologies. Similarly, Moreira et al. [24] proposed an RL solution for multi-domain and complex infrastructures. In [25], the authors proposed a heuristic to solve three formulated problems for multi-cast SFC, readjustment, and expanding to cover the provisioning while considering service migration. However, these approaches fail to adequately capture the impact of isolation due to the overheads and usage patterns across isolation levels, which can significantly impact performance. In addition, most existing RL algorithms require substantial training iterations, and they are unreliable in unseen situations, undermining their practicality for HP SRs. On the other hand, heuristics often demonstrate sub-optimal performance.

2) *Isolation Concept*: In [20], the authors classified computing isolation methods into language-based, sandbox, VM-based, OS-kernel, and hardware-based techniques. However, a 5G InP might not be able to utilize all these methods. The authors then suggested E2E isolation levels using a weighted power mean function, which does not necessarily yield a meaningful security metric. The authors in [13] defined isolation levels for 5G resources based on their shared portions. They described isolation approaches for radio, transport, and MEC/cloud-based on the latest standards and technologies. However, these studies focused on isolation as a sole concept in 5G and did not evaluate its impact on NS.

3) *Isolation-Aware NS*: A few works studied integrating isolation approaches in NS. The authors in [11] proposed a solution for performing isolation-aware 5G RAN slicing enabled by wavelength division multiplexing (WDM). Nevertheless, isolation is enforced at the network domain level. Thus, isolated domains can still be vulnerable depending on their resource isolation level. In [22], the authors introduced an RL-based approach to improve slice isolation against attacks. Their solution involved finding an efficient distribution of slices' resources to mitigate the impact of DDoS attacks on a slice from affecting others. Despite overall improvements, their solution does not eliminate the effect for HP SRs and merely considers the servers with small topologies. Some works focus on another isolation dimension, namely performance isolation [10], [15]. Yarkina et al. in [10] proposed a priority-based performance isolation for 5G radio access network (RAN) and assessed some standard-compliant machine learning (ML) approaches. According to [10], ML approaches are prone to service requirement violations under high-load conditions. Cheng et al. [15] proposed a slice-aware performance isolation solution for open RAN based on open-source solutions. Although this work can be seen as a proof-of-concept, it does not cover security aspects of isolation or aim to study isolation at levels based on recommendations.

A summarized comparison of the novelties of this work in five key attributes versus existing slice isolation approaches is given in Table I. The comparison is structured along five key attributes: 1) the granularity of isolation (i.e., domain-specific, DDoS-focused, or E2E), 2) the targeted isolation dimension (e.g., security or performance), 3) whether slice priority is considered during embedding, 4) whether isolation-induced overheads are

TABLE I  
SUMMARY OF EXISTING SLICE ISOLATION APPROACHES VERSUS 5GUARD

Work	Isolation Granularity	Isolation Dimension	Priority-aware	Overhead Analysis	NS Scope
[13], [20]	Domain	Security	✗	✗	✗
[16], [17]	Domain	Security	✗	✗	E2E
[21], [22]	DDoS	Security	✗	✗	5GC
[10], [15]	✗	Performance	✓	✗	RAN
<b>This work</b>	E2E	Security	✓	✓	E2E

analyzed and incorporated into resource allocation, and 5) the scope of NS (e.g., RAN, E2E, or no NS).

### III. SYSTEM MODEL

In this section, we present our 5G network and isolation models, detailing isolation levels with use cases, and introduce the SR model.

#### A. Isolation and Network Model

As illustrated in Fig. 1, we consider a 5G infrastructure, consisting of new radio base stations (NR-BS), MEC, and telco cloud servers, as well as single/multi-hop optical links. Without loss of generality, and based on the recommendations in [9], [30], we define the following three E2E isolation levels:

- **Complete isolation (L2)**: This level can support mission-critical (MC) slices, such as emergency [31], public [32] and maritime [33] safety, e-health, government, and military services, as well as virtual operators [34]. We consider dedicated NR-BSs, fiber links, and physical MEC/cloud server nodes.
- **Semi isolation (L1)**: This level can benefit services that require isolation but cannot afford the high costs of complete isolation, such as ultra-reliable low latency communication (uRLLC) services and 5G-V2X [35]. NR-BSs are shared while dedicating physical resource blocks (PRBs) for the entire radio frame to an SR [13] with additional guard bands, as defined in standards [36]. For fiber optic links, WDM or elastic optical network (EON) can be used with dedicated wavelength subcarriers and standard guard bands [37]. Finally, a VM is instantiated for each network function (NF) instance.
- **Minimum/no isolation (L0)**: This level benefits basic SRs for best-effort 5G services, for instance. PRBs are dedicated to a fraction of the radio frame based on orthogonal frequency-division multiple access (OFDMA), as defined in the standards [36]. Tagging approaches, e.g., multiprotocol label switching (MPLS), can be leveraged to differentiate between slices on the transport while the bandwidth subcarriers are shared. NFs are deployed as containers sharing VMs. We also define the VNF-sharing sub-level, indicating whether containers are shareable. By consolidating SRs in shared entities, i.e., containers, InPs can enhance resource utilization while accommodating more L2 requests.

One note to consider in defining these isolation levels is that specifications have defined standardized slice/service types

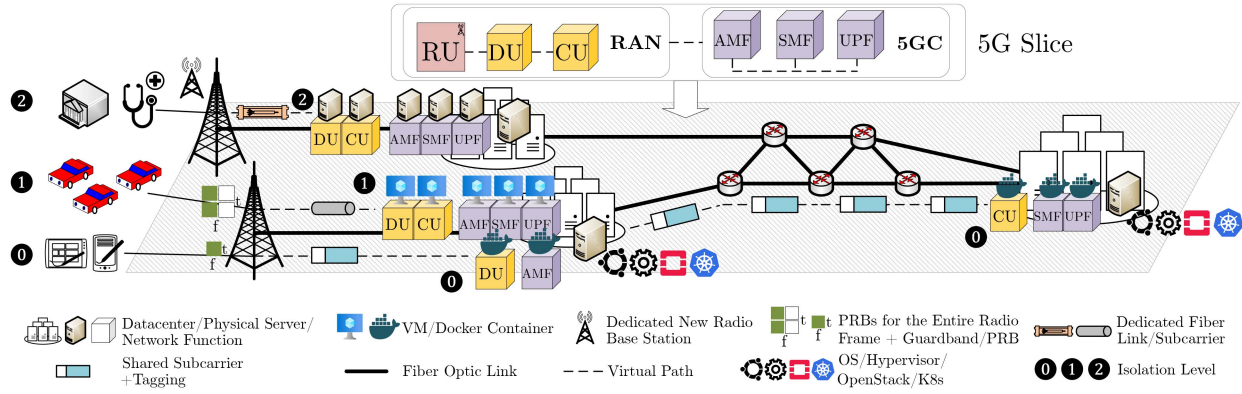


Fig. 1. The proposed isolation-aware 5G network slicing system model. On top, an E2E 5G slice spanning from the RAN to the 5GC is illustrated, with slice components—NFs, RUs, and VPs—deployable on resources with three isolation levels: L2, L1, and L0. The left-hand side demonstrates three users of 5G slices: L2 for government and medical services, L1 for 5G-V2X services, and L0 for public streaming and entertainment services. Notably, the L2 slice is fully isolated, sharing no resources with other slices. In the example shown, servers' revisitation increases resource efficiency.

TABLE II  
KEY NOTATIONS

	Symbol	Description
Variables	$x_v^p$	Mapping of NF $v$ to PS $p$
	$y_w^{q,\sigma,k}$	Assigning $k$ PRBs to RU $w$ from frequency slot $\sigma$ of NR-BS $q$
	$z_{vw}^{l,\lambda}$	Using wavelength slot $\lambda$ for VP $e_{vw}$ from link $l$
	$c_v^p$	MIPS capacity of NF $v$ from PS $p$
	$b_{vw}^{pq,i,l,\lambda}$	Bandwidth of VP $e_{vw}$ from PP $e_{pq}^{(i)}$ using wavelength slot $\lambda$ of link $l$
Sets	$\mathcal{R}, \mathcal{R}_t^{\text{HP}}, \mathcal{R}_t^{\text{LP}}$	Set of all SRs, excluding pre-embedded HP SRs, and pre-embedded HP and LP SRs at time $t$
	$\mathcal{V}_r^{\text{S}}, \mathcal{V}_r^{\text{R}}, \mathcal{E}_r$	Set of NF and RU nodes and VPs of SR $r$
	$\mathcal{V}^{\text{PS}}, \mathcal{V}^{\text{PR}}, \mathcal{E}^{\text{P}}$	Set of PS and NR-BS nodes and PPs
	$\mathcal{L}^{\text{P}}, \mathcal{L}_{pq}^{\text{P}}$	Set of all links and links composing PP $e_{pq}^{(i)}$
Parameters	$\gamma_r, K_r$	E2E isolation level and sub-level for VNF sharing
	$C_v^{\text{REQ}}, K_w^{\text{REQ}}, B_{vw}^{\text{REQ}}, D_{vw}^{\text{REQ}}$	Requested MIPS and PRBs of NF $v$ and RU $w$ , and bandwidth and delay of VP $e_{vw}$
	$\Xi_v^{\text{REQ}}, \chi_{v,\xi}^{\text{REQ}}$	Requested number of VNF instances and VNF types of NF $v$
	$R_r, \rho_r, P^{\text{min}}$	SR $r$ 's revenue and QoS violation cost rate vector and minimum acceptable increased InP cumulative profit from $\mathcal{R}$
	$\Omega, \Phi, \Theta, \beta_r$	MIPS, PRB, and bandwidth cost rate for PSs, RUs, and links, and migration cost rate of SR $r$
	$\Pi_{q,\sigma}^{\text{max}}, B_{l,\lambda}^{\text{max}}$	Maximum radio and bandwidth capacity of NR-BS $q$ on frequency slot $\sigma$ and link $l$ on wavelength slot $\lambda$
	$\Pi^{\text{PRB}}, N^{\text{FRM}}$	Radio resources of a PRB and the number of PRBs in a radio frame
	$C_{\text{HHO}}, C_{\text{K8S}}, C_{\text{GOS}}, \Pi_{\text{FGB}}$	MIPS overhead of the hypervisor, host OS, and OpenStack, K8s, and guest OS and radio overhead of a guard band
$C_p^{\text{max}}, \Xi_{p,\gamma}^{\text{max}}, D_{pq}^{\text{RTT}}$	Maximum MIPS and instances at PS $p$ and isolation level $\gamma$ and round-trip-time delay of PP $e_{pq}^{(i)}$	

(SSTs), e.g., enhanced mobile broadband (eMBB), uRLLC, and massive machine type communication (mMTC), based on QoS requirements, such as bandwidth and latency [31]. However, SSTs do not capture the security or isolation requirements of slices. There, we define isolation levels as an orthogonal requirement to SSTs. Any SST can be deployed with any isolation level, depending on the tenant's SLA. For instance, a uRLLC slice for autonomous driving may require an L2 isolation level in an MC context but L1 in a commercial one. Therefore, rather than mapping SSTs to fixed isolation levels, we provided use case examples from standards that may benefit from each level, reflecting realistic requirements.

Next, we model the 5G substrate network as an undirected graph with labeled nodes and edges denoted by  $\mathcal{G}^{\text{P}} \triangleq (\mathcal{V}^{\text{PR}}, \mathcal{V}^{\text{PS}}, \mathcal{E}^{\text{P}})$ , where  $\mathcal{V}^{\text{PR}}$  denotes the set of NR-BS nodes,  $\mathcal{V}^{\text{PS}}$  denotes the set of MEC/cloud physical server (PS) nodes, and  $\mathcal{E}^{\text{P}}$  denotes the set of physical paths (PPs) between nodes. PPs are composed of a set of interconnected links, denoted by  $\mathcal{L}^{\text{P}}$ . PS and

NR-BS node labels reflect the radio and computing resources' capacity, expressed as a product of a unit of frequency and time and a million instructions per second (MIPS), respectively. PS node labels also represent the maximum number of instances at each isolation level (e.g., VMs or containers). The goal of defining such a limit is to cap the overhead of the hypervisor, guest/host OS, and container orchestration tool (e.g., K8s). PP labels indicate the round-trip-time (RTT) delay and the link bandwidth resources. The key notations and acronyms are listed in Tables II and III, respectively.

### B. Slice Request (SR) Model

The 5G infrastructure described earlier supports multiple slices with different isolation levels, where each slice has a service-level agreement (SLA). We assume that SRs arrive in an online setting and are stored in an ingress first-in-first-out queue. Based on the SLA, we model an SR, as tuple

TABLE III  
KEY ACRONYMS

Acronym	Definition	Acronym	Definition
DU/CU	Distributed/Centralized Unit	E2E	End-to-End
InP	Infrastructure Provider	K8s	Kubernetes
L0/L1/L2	Isolation Level 0/1/2	LP/HP	Low/High Priority
LP/ILP	Linear/Integer Linear Programming	MC	Mission-Critical
MIPS	Million Instructions per Second	NF	Network Function
NR-BS	New Radio Base Station	NS	Network Slicing
OS	Operating System	PN	Physical Network
PRB	Physical Resource Block	PS	Physical Server
QoS	Quality of Service	RU	Radio Unit
SLA	Service-Level Agreement	SR	Slice Request
VN/VM	Virtual Network/Machine	VP/PP	Virtual/Physical Path

$r \triangleq (\mathcal{V}_r^R, \mathcal{V}_r^S, \mathcal{E}_r, \gamma_r, \kappa_r, R_r, \beta_r, T_r, \rho_r)$ , where  $\mathcal{V}_r^R$ ,  $\mathcal{V}_r^S$ , and  $\mathcal{E}_r$  denote the set of radio unit (RU) and NF nodes and virtual paths (VPs) of the VN, respectively.  $\gamma_r \in \{0, 1, 2\}$  and  $\kappa_r \in \{0, 1\}$  denote the isolation level and L0 VNF sharing sub-level, respectively.  $R_r \in \mathbb{R}_{>0}$  denotes the revenue,  $\beta_r \in \mathbb{R}_{\geq 0}$  denotes the cost of a unit of migration,  $T_r \in \mathbb{R}_{>0}$  denotes the resource allocation deadline,  $\rho_r \in \mathbb{R}_{\geq 0}^3$  is the SR's cost vector of a unit of QoS violation for radio, MIPS, and bandwidth resources. For recurring slices, the customer must request a new slice at the end of the slice's lifetime. The above modeling approach for slices as graphs is compliant with 3rd Generation Partnership Project (3GPP) TS 28.530 [8], the QoS parameters align with the service requirements specified in 3GPP TS 23.501 [31], and the cost vectors are design choices typically determined by the business models of InPs.

#### IV. 5G-INS FORMULATION

In this section, we formally define 5G-INS as the 5G isolation-aware network slicing problem, formulating it through constraints and an objective function. We also analyze its complexity and reduce it to a manageable form.

##### A. Problem

We explore the 5G-INS problem from the InP's perspective, responsible for making slicing decisions to ensure customers receive services aligned with their SLAs. The goal is to embed each incoming SR, denoted by  $r_t$ , upon arrival at time step  $t \in \mathbb{Z}_{\geq 0}$ , and reconfigure the pre-embedded LP slices in the set denoted by  $\mathcal{R}_t^{\text{LP}}$  as needed. This should consider the slices' priority and associated costs to meet isolation level  $\gamma_r$  and the QoS requirements for all SRs. We assume the InP aims to maximize profit and must determine how to map VNs best and allocate resources on PN  $\mathcal{G}^P$ . In addition, InP should ensure that it does not exceed the available resources while factoring in the overheads and resource usage patterns of isolation levels. Considering these constraints, we assume that InP intends to admit all SRs arriving at different time steps, unless infeasible.

##### B. Decision Variables

Let  $\mathcal{R} \triangleq \mathcal{R}_t^{\text{LP}} \cup \{r_t\}$  denote the set of all SRs, except the pre-embedded HP SRs, at time step  $t$ . For mapping NFs to PSs,

we define binary variable  $x_v^p$  in vector  $\mathbf{x}$  that indicates whether NF  $v \in \mathcal{V}_r$  of SR  $r \in \mathcal{R}$  is mapped to PS  $p \in \mathcal{V}^P$ . We also denote  $c_v^p \in \mathbb{R}_{\geq 0}$  in vector  $\mathbf{c}$  as the amount of MIPS allocated to NF  $v \in \mathcal{V}_r$  of SR  $r \in \mathcal{R}$  from PS  $p \in \mathcal{V}^P$ . Since radio resources are typically allocated in PRBs, they are inherently integer. Additionally, resources are released at the end of a slice's lifetime. Thus, considering pre-embedded HP SRs' allocated resources, they can be fragmented into a set of frequency slots denoted by  $\mathcal{S}_q$  for NR-BS  $q \in \mathcal{V}^{\text{PR}}$ . Thus, we define binary variable  $y_w^{q,\sigma,k}$  in vector  $\mathbf{y}$  indicating whether  $k$  PRBs are allocated to RU  $w \in \mathcal{V}_r^R$  of SR  $r \in \mathcal{R}$  from the frequency slot  $\sigma \in \mathcal{S}_q$  of NR-BS  $q \in \mathcal{V}^{\text{PR}}$ .

Bandwidth fragmentation can similarly occur for fiber optic links. Thus, we denote  $\mathcal{D}_l$  as the set of wavelength slots for link  $l \in \mathcal{L}^P$ . We define binary variable  $z_{vw}^{l,\lambda}$  in vector  $\mathbf{z}$  that indicates whether VP  $e_{vw} \in \mathcal{E}_r$ , the VP connecting NF/RU nodes  $v \in \mathcal{V}_r^S \cup \mathcal{V}_r^R$  and  $w \in \mathcal{V}_r^S \cup \mathcal{V}_r^R$  of SR  $r \in \mathcal{R}$ , uses the wavelength slot  $\lambda \in \mathcal{D}_l$  of link  $l \in \mathcal{L}^P$ . To avoid bottlenecks and increase utilization, we allow bandwidth splitting between the links of multiple PPs. Thus, we define variable  $b_{vw}^{pqi,l,\lambda} \in \mathbb{R}_{\geq 0}$  in vector  $\mathbf{b}$  as the bandwidth allocated to VP  $e_{vw} \in \mathcal{E}_r$  of SR  $r \in \mathcal{R}$  from slot  $\lambda \in \mathcal{D}_l$  of link  $l \in \mathcal{L}^P$ , where  $\mathcal{L}^{pqi}$  denotes the set of links on PP  $e_{pq}^{(i)} \in \mathcal{E}^P$ .

##### C. NF and RU Mapping Constraints

1) *Mapping Uniqueness*: To ensure each NF and RU is precisely mapped to a single PS and NR-BS, respectively, we enforce,

$$\sum_{p \in \mathcal{V}^{\text{PS}}} x_v^p = 1, \quad \forall v \in \mathcal{V}_r^S, r \in \mathcal{R}, \quad (1)$$

$$\sum_{q \in \mathcal{V}^{\text{PR}}} \sum_{\sigma \in \mathcal{S}_q} \sum_{k=1}^{K_w^{\text{REQ}}} y_w^{q,\sigma,k} = 1, \quad \forall w \in \mathcal{V}_r^R, r \in \mathcal{R} \quad (2)$$

where  $K_w^{\text{REQ}} \in \mathbb{Z}_{>0}$  denotes the number of PRBs requested by RU  $w \in \mathcal{V}_r^R$  of SR  $r \in \mathcal{R}$ . For RUs, a single frequency slot can be selected in the mapped NR-BS to maintain spectral contiguity. The summation over  $k$  prevents selecting multiple values for the assigned PRBs. Note that for pre-embedded HP SRs, we reuse the previous NF and RU mappings, since migration can lead to service degradation.

2) *Complete Isolation Mapping*: For ease of exposition, we first define  $f_{\hat{p}}(\hat{\mathcal{V}}, \hat{\mathcal{R}}, \mathbf{a}^{\hat{p}})$  as the function that checks how many L2 SRs have at least one virtual entity, i.e., NF, RU, or VP, in set  $\hat{\mathcal{V}} \triangleq \bigcup_{r \in \hat{\mathcal{R}}}^{\gamma_r=2} \hat{\mathcal{V}}_r$  that are mapped to physical entity  $\hat{p}$ , i.e., a PS, NR-BS, or PP, based on the mapping decision vector  $\mathbf{a}^{\hat{p}} \triangleq (a_{\hat{v}}^{\hat{p}})_{\hat{v} \in \hat{\mathcal{V}}}$ . Thus,

$$f_{\hat{p}}(\hat{\mathcal{V}}, \hat{\mathcal{R}}, \mathbf{a}^{\hat{p}}) \triangleq \sum_{r \in \hat{\mathcal{R}}}^{\gamma_r=2} \max_{\hat{v} \in \hat{\mathcal{V}}_r} a_{\hat{v}}^{\hat{p}} \quad (3)$$

For L2, a PS or NR-BS must be mapped to at most one NF or RU, respectively. Accordingly, we need to enforce the following L2 constraints for each PS and NR-BS, respectively.

$$f_p(\mathcal{V}^S, \mathcal{R}, \mathbf{x}^p) \leq 1, \quad \forall p \in \mathcal{V}^{\text{PS}} \quad (4)$$

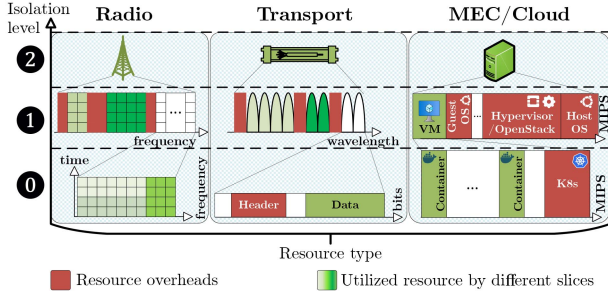


Fig. 2. The proposed isolation levels and resource overheads: For radio resources, these include dedicated NR-BS, PRBs for the entire radio frame with guard bands, and PRBs allocated via OFDMA. Transport isolation uses dedicated fiber, subcarriers with WDM/EON, and shared subcarriers with tagging. MEC/cloud employs servers, VMs, and containerized K8s pods.

where  $\mathbf{x}^p \triangleq (x_v^p)_{v \in \mathcal{V}_r^S}^{r \in \mathcal{R}}$  and  $\mathcal{V}^S \triangleq \bigcup_{r \in \mathcal{R}} \mathcal{V}_r^S$ .

$$f_q(\mathcal{V}^R, \mathcal{R}, \mathbf{y}^q) \leq 1, \quad \forall q \in \mathcal{V}^{\text{PR}} \quad (5)$$

where  $\mathbf{y}^q \triangleq (\sum_{\sigma \in \mathcal{S}_q} \sum_{k=1}^{K_w^{\text{REQ}}} y_w^{q,\sigma,k})_{w \in \mathcal{V}_r^R}$  and  $\mathcal{V}^R \triangleq \bigcup_{r \in \mathcal{R}} \mathcal{V}_r^R$ .

To prevent other slices from being mapped to the L2 slices' PSs and NR-BSs,

$$x_v^p \leq 1 - f_p(\mathcal{V}^S, \mathcal{R} \setminus \{r\}, \mathbf{x}^p), \quad \forall v \in \mathcal{V}_r^S, r \in \mathcal{R}, p \in \mathcal{V}^{\text{PS}} \quad (6)$$

$$\sum_{\sigma \in \mathcal{S}_q} \sum_{k=1}^{K_w^{\text{REQ}}} y_w^{q,\sigma,k} \leq 1 - f_q(\mathcal{V}^R, \mathcal{R} \setminus \{r\}, \mathbf{y}^q), \quad \forall w \in \mathcal{V}_r^R, r \in \mathcal{R}, q \in \mathcal{V}^{\text{PR}} \quad (7)$$

3) *Coordination and Overprovisioning Prevention*: When an NF is not mapped to a PS, the allocated capacity associated with that mapping should also be zero, and vice versa. Additionally, to prevent overprovisioning,

$$\delta x_v^p \leq c_v^p \leq C_v^{\text{REQ}} x_v^p, \quad \forall p \in \mathcal{V}^{\text{PS}}, v \in \mathcal{V}_r^S, r \in \mathcal{R} \quad (8)$$

where  $C_v^{\text{REQ}} \in \mathbb{R}_{>0}$  and  $\delta \in \mathbb{R}_{>0}$  are MIPS requested by NF  $v \in \mathcal{V}_r^S$  of SR  $r \in \mathcal{R}$  and the smallest MIPS capacity that can be allocated on PSs, respectively. Similarly, for radio resources,

$$\sum_{\sigma \in \mathcal{S}_q} \sum_{k=1}^{K_w^{\text{REQ}}} k y_w^{q,\sigma,k} \leq K_w^{\text{REQ}}, \quad \forall q \in \mathcal{V}^{\text{PR}}, w \in \mathcal{V}_r^R, r \in \mathcal{R} \quad (9)$$

4) *Maximum Capacity*: The allocated MIPS to each PS across all isolation levels should not exceed its capacity. As mentioned earlier, isolation levels contribute to specific MIPS overheads, either from the isolation itself or are required for operation at that isolation level. These overheads arise from OpenStack, a host OS, hypervisor, and several guest OSs and container orchestration tools, i.e., K8s, which scale with the number of VMs.

To calculate these, we first need to derive the number of VMs for L0 SRs. According to Fig. 2, this controls how much overhead is imposed by the container orchestration tool (e.g., K8s) and guest OSs. This paper assumes containers are placed

in VMs to improve isolation from L1. This is also common in creating K8s clusters, the substrate to run the containers. For ease of exposition, we first define  $h_p^{\text{L0,0,U}}$ ,  $h_p^{\text{L0,0,S}}$  as the number of VMs of L0 SRs using PS  $p \in \mathcal{V}^{\text{PS}}$  with isolation sub-level of 0 for two types of VNFs: 1) unsharable and 2) sharable, respectively as follows,

$$h_p^{\text{L0,0,U}} \triangleq \sum_{r \in \mathcal{R} \cup \mathcal{R}_t^{\text{HP}}} \sum_{v \in \mathcal{V}_r^S} \sum_{\xi=1}^{\Xi_v^{\text{REQ}}} \frac{\tilde{x}_v^p}{\Xi_{p,0}^{\text{max}}} \quad (10)$$

$$h_p^{\text{L0,0,S}} \triangleq \sum_{\chi=0}^{\chi_{\text{max}}^S} \max_{v \in \mathcal{V}_r^S, 1 \leq \xi \leq \Xi_v^{\text{REQ}}, \gamma_r = \kappa_r = 0, r \in \mathcal{R} \cup \mathcal{R}_t^{\text{HP}}} \frac{\mathbb{1}_{\{\chi_{v,\xi}^{\text{REQ}} = \chi\}} \tilde{x}_v^p}{\Xi_{p,0}^{\text{max}}} \quad (11)$$

where

$$\tilde{x}_v^p \triangleq \begin{cases} x_v^p, & \forall v \in \mathcal{V}_r^S, r \in \mathcal{R} \\ X_v^p, & \forall v \in \mathcal{V}_r^S, r \in \mathcal{R}_t^{\text{HP}} \end{cases}$$

with  $X_v^p \in \{0, 1\}$  denoting the existing NF mapping values for pre-embedded HP SRs. In addition,  $\mathbb{1}_{\{\cdot\}}$  is the indicator function, and  $\Xi_{p,0}^{\text{max}} \in \mathbb{Z}_{>0}$  represents the maximum number of containers recommended by the orchestration tool to avoid resource contention. Here,  $\chi_{v,\xi}^{\text{REQ}}$  denotes the type of VNF instance  $\xi \in [1, \Xi_v^{\text{REQ}}]$  of NF  $v \in \mathcal{V}_r^S$  requested by SR  $r \in \mathcal{R}$ , with  $\Xi_v^{\text{REQ}} \in \mathbb{Z}_{>0}$  denoting the number of requested VNF instances. In addition,  $\chi^U \in \mathbb{Z}_{\geq 0}$  and  $\chi_{\text{max}}^S \in \mathbb{Z}_{>0}$  denote the unsharable VNF type and the maximum number of sharable types, respectively. The max function used in (11) ensures that one container is instantiated for an individual sharable type at most. Then, by applying a division by  $\Xi_{p,0}^{\text{max}}$ , the number of VMs is derived.

Similarly, we define  $h_p^{\text{L0,1}}$  as the number of VMs of L0 SRs with isolation sub-level of 1, where no VNF sharing is allowed, as follows.

$$h_p^{\text{L0,1}} \triangleq \sum_{r \in \mathcal{R} \cup \mathcal{R}_t^{\text{HP}}} \sum_{v \in \mathcal{V}_r^S} \frac{\Xi_v^{\text{REQ}} \tilde{x}_v^p}{\Xi_{p,0}^{\text{max}}}, \quad (12)$$

Here, a container is instantiated for each requested instance, similar to (10).

Accordingly, we can derive the overall number of VMs for L0 SRs, denoted by  $h_p^{\text{L0}}$ , as follows.

$$h_p^{\text{L0}} = \lceil h_p^{\text{L0,0,U}} + h_p^{\text{L0,0,S}} + h_p^{\text{L0,1}} \rceil, \quad (13)$$

where  $\lceil \cdot \rceil$  avoids deriving a fractional number.

By utilizing (13), we can derive the total MIPS overhead of deploying the isolation levels for PS  $p \in \mathcal{V}^{\text{PS}}$ , denoted by  $C_{p,\text{TOT}}^{\text{OVH}}$ , as follows.

$$C_{p,\text{TOT}}^{\text{OVH}} = (C_{\text{K8S}} + C_{\text{GOS}}) h_p^{\text{L0}} + C_{\text{HHO}} \max_{v \in \mathcal{V}_r^S, \gamma_r \leq 1, r \in \mathcal{R} \cup \mathcal{R}_t^{\text{HP}}} \tilde{x}_v^p + C_{\text{GOS}} \sum_{r \in \mathcal{R} \cup \mathcal{R}_t^{\text{HP}}} \sum_{v \in \mathcal{V}_r^S} \Xi_v^{\text{REQ}} \tilde{x}_v^p \quad (14)$$

where  $C_{\text{K8S}} \in \mathbb{R}_{>0}$ ,  $C_{\text{GOS}} \in \mathbb{R}_{>0}$ , and  $C_{\text{HHO}} \in \mathbb{R}_{>0}$  are the maximum MIPS overhead of the container orchestration tool

(i.e., K8s), guest OS, and hypervisor, host OS, and OpenStack, respectively. Note that the MIPS required for making radio resource allocation decisions is assumed to run on separate management servers and does not consume slice-assigned resources. In contrast, RAN-related MIPS, e.g., for distributed units (DUs), is included in the SR and accounted for in the consumed MIPS vector  $\mathbf{c}$ , and not as an overhead.

Now that we have derived the total MIPS overhead, by utilizing (14), the constraint can be determined as follows.

$$\sum_{r \in \mathcal{R} \cup \mathcal{R}_t^{\text{HP}}} \sum_{v \in \mathcal{V}_r^{\text{S}}} \tilde{c}_v^p \leq C_p^{\text{max}} - C_{p,\text{TOT}}^{\text{OVH}}, \quad \forall p \in \mathcal{V}^{\text{PS}} \quad (15)$$

where

$$\tilde{c}_v^p \triangleq \begin{cases} c_v^p, & \forall v \in \mathcal{V}_r^{\text{S}}, r \in \mathcal{R} \\ C_v^p, & \forall v \in \mathcal{V}_r^{\text{S}}, r \in \mathcal{R}_t^{\text{HP}} \end{cases}$$

with  $C_v^p \in \mathbb{R}_{\geq 0}$  denoting the existing allocated MIPS capacity to  $v \in \mathcal{V}_r^{\text{S}}$  for pre-embedded HP SR  $r \in \mathcal{R}_t^{\text{HP}}$  and  $C_p^{\text{max}}$  is the MIPS capacity of PS  $p \in \mathcal{V}^{\text{PS}}$ .

Considering the VMs instantiated for slices at L1 and for hosting containers at L0, there should be a limit on the number of VMs on PS  $p \in \mathcal{V}^{\text{PS}}$ , denoted by  $\Xi_{p,1}^{\text{max}} \in \mathbb{Z}_{>0}$ . Otherwise, the MIPS capacity overhead of the hypervisor can grow significantly by constantly adding VMs. Therefore,

$$h_p^{\text{L0}} + \sum_{r \in \mathcal{R} \cup \mathcal{R}_t^{\text{HP}}} \sum_{v \in \mathcal{V}_r^{\text{S}}} \Xi_v^{\text{REQ}} \tilde{x}_v^p \leq \Xi_{p,1}^{\text{max}}, \quad \forall p \in \mathcal{V}^{\text{PS}} \quad (16)$$

For each NR-BS, we should not surpass its maximum radio resources at each frequency slot, considering the coexistence of L0 and L1 and the frequency guard bands' overhead. First, let  $\Pi_{\text{FGB}}^{\text{OVH}}$  denote the radio overhead imposed by each frequency guard band at L1. We can then derive the total utilized radio and radio overhead for frequency slot  $\sigma \in \mathcal{S}_q$  of NR-BS  $q \in \mathcal{V}^{\text{PR}}$ , denoted by  $\Pi_{q,\text{TOT}}^{\sigma,\text{UTL}}$  and  $\Pi_{q,\text{TOT}}^{\sigma,\text{OVH}}$ , respectively, as follows.

$$\begin{aligned} \Pi_{q,\text{TOT}}^{\sigma,\text{UTL}} &= \Pi^{\text{PRB}} \left( \sum_{r \in \mathcal{R} \cup \mathcal{R}_t^{\text{HP}}} \sum_{w \in \mathcal{V}_r^{\text{R}}} \sum_{k=1}^{K_w^{\text{REQ}}} k \tilde{y}_w^{q,\sigma,k} \right. \\ &\quad \left. + N^{\text{FRM}} \sum_{r \in \mathcal{R} \cup \mathcal{R}_t^{\text{HP}}} \sum_{w \in \mathcal{V}_r^{\text{R}}} \sum_{k=1}^{K_w^{\text{REQ}}} \left\lceil \frac{k \tilde{y}_w^{q,\sigma,k}}{N^{\text{FRM}}} \right\rceil \right) \quad (17) \end{aligned}$$

$$\Pi_{q,\text{TOT}}^{\sigma,\text{OVH}} = \sum_{r \in \mathcal{R} \cup \mathcal{R}_t^{\text{HP}}} \sum_{w \in \mathcal{V}_r^{\text{R}}} \sum_{k=1}^{K_w^{\text{REQ}}} \tilde{y}_w^{q,\sigma,k} \Pi_{\text{FGB}}^{\text{OVH}} \quad (18)$$

where

$$\tilde{y}_w^{q,\sigma,k} \triangleq \begin{cases} y_w^{q,\sigma,k}, & \forall w \in \mathcal{V}_r^{\text{R}}, r \in \mathcal{R} \\ Y_w^{q,\sigma,k}, & \forall w \in \mathcal{V}_r^{\text{R}}, r \in \mathcal{R}_t^{\text{HP}} \end{cases}$$

with  $Y_w^{q,\sigma,k} \in \{0, 1\}$  denoting the existing RU mapping values for pre-embedded HP SRs. In addition,  $N^{\text{FRM}} \in \mathbb{Z}_{>0}$ , and  $\Pi^{\text{PRB}} \in \mathbb{R}_{>0}$  denote the number of PRBs in a radio frame and the radio resources occupied by a single PRB, respectively. Overall, the ceiling function  $\lceil \cdot \rceil$  restricts L1 RUs only to be assigned to full radio frames and not the fraction of the frame that may be unused by L0.

Thus, using (17) and (18), we should enforce the following constraint.

$$\Pi_{q,\text{TOT}}^{\sigma,\text{UTL}} \leq \Pi_{q,\sigma}^{\text{max}} - \Pi_{q,\text{TOT}}^{\sigma,\text{OVH}}, \quad \forall \sigma \in \mathcal{S}_q, q \in \mathcal{V}^{\text{PR}} \quad (19)$$

where  $\Pi_{q,\sigma}^{\text{max}}$  denote the maximum radio resources at frequency slot  $\sigma \in \mathcal{S}_q$  of NR-BS  $q \in \mathcal{V}^{\text{PR}}$ .

#### D. VP Mapping Constraints

1) *Mapping Uniqueness*: In this paper, we assume each VP is mapped to at most one PP to maintain spectral contiguity by ensuring a contiguous bandwidth allocation and preventing fragmentation. This approach also reduces overhead for L1 SRs, as bandwidth splitting would require dedicated guardbands and subcarrier for each VP. Thus,

$$\sum_{e_{pq}^{(i)} \in \mathcal{E}^{\text{P}}} \min_{l \in \mathcal{L}_{pq}^i} \sum_{\lambda \in \mathcal{D}_l} z_{vw}^{l,\lambda} \leq 1, \quad \forall e_{vw} \in \mathcal{E}_r, r \in \mathcal{R} \quad (20)$$

where the inequality accounts for the fact that no mapping is required for revisited VPs. In addition,  $\min(\cdot)$  selects the paths with all the constituent links being mapped. The summation over the set of wavelength slots of link  $l \in \mathcal{L}^{\text{P}}$  ensures the spectral contiguity by selecting at most a single slot  $\lambda \in \mathcal{D}_l$  on  $l$  for VP  $e_{vw} \in \mathcal{E}_r$  of SR  $r \in \mathcal{R}$ .

2) *Complete Isolation Mapping*: At L2, a link is only mapped to at most a single SR. Thus,

$$f_l(\mathcal{E}, \mathcal{R}, \mathbf{z}^l) \leq 1, \quad \forall l \in \mathcal{L}^{\text{P}} \quad (21)$$

$$\begin{aligned} \sum_{\lambda \in \mathcal{D}_l} z_{vw}^{l,\lambda} &\leq 1 - f_l(\mathcal{E}, \mathcal{R} \setminus \{r\}, \mathbf{z}^l), \\ \forall l \in \mathcal{L}^{\text{P}}, e_{vw} &\in \mathcal{E}_r, r \in \mathcal{R} \end{aligned} \quad (22)$$

where  $\mathbf{z}^l \triangleq (\sum_{\lambda \in \mathcal{D}_l} z_{vw}^{l,\lambda})_{e_{vw} \in \mathcal{E}_r, r \in \mathcal{R}}$ .

3) *Coordination and Overprovisioning Prevention*: As previously mentioned, we employ tagging for L0 service differentiation, with a fixed-sized header in packets to distinguish slices. Thus, they consume a specific percentage of the allocated bandwidth, denoted by  $\psi_{\text{HDR}}^{\text{OVH}} \in \mathbb{R}_{>0}$ , which is essential for L0 operation. Given this overhead, we must ensure coordinated bandwidth allocations and mapping decisions. In addition, we should prevent overprovisioning. Thus,

$$\begin{aligned} \epsilon z_{vw}^{l,\lambda} &\leq \sum_{e_{pq}^{(i)} \in \mathcal{E}_l^{\text{P}}} b_{vw}^{pq,i,l,\lambda} \leq \frac{B_{vw}^{\text{REQ}} z_{vw}^{l,\lambda}}{1 - \psi_{\text{HDR}}^{\text{OVH}} \mathbb{1}_{\{\gamma_r=0\}}}, \\ \forall l \in \mathcal{L}^{\text{P}}, \lambda \in \mathcal{D}_l, e_{vw} &\in \mathcal{E}_r, r \in \mathcal{R} \end{aligned} \quad (23)$$

where  $\epsilon \in \mathbb{R}_{>0}$  is the smallest allocatable bandwidth. A PP's bandwidth is limited by its bottleneck link. Therefore, to efficiently consume bandwidth resources, we ensure that links on a PP receive the same bandwidth, considering their potentially different bandwidth fragmentation as follows.

$$\begin{aligned} \sum_{\lambda \in \mathcal{D}_{l'}} b_{vw}^{pq,i,l',\lambda} &= \sum_{\lambda \in \mathcal{D}_{l''}} b_{vw}^{pq,i,l'',\lambda}, \\ \forall l', l'' \in \mathcal{L}_{pq}^i, l' \neq l'', e_{pq}^{(i)} &\in \mathcal{E}^{\text{P}}, e_{vw} \in \mathcal{E}_r, r \in \mathcal{R} \end{aligned} \quad (24)$$

It is essential that if a VP is mapped to a PP, its end nodes are also mapped. Accordingly,

$$\sum_{e_{pq}^{(i)} \in \mathcal{E}^P} \sum_{\lambda \in \mathcal{D}_{l_1}} b_{vw}^{pqi,l_1,\lambda} \leq \frac{B_{vw}^{\text{REQ}}}{1 - \psi_{\text{HDR}}^{\text{OVH}} \mathbb{1}_{\{\gamma_r=0\}}} x_v^p, \quad \forall e_{vw} \in \mathcal{E}_r, p \in \mathcal{V}^{\text{PS}}, v \in \mathcal{V}_r^{\text{S}}, r \in \mathcal{R} \quad (25)$$

$$\sum_{e_{pq}^{(i)} \in \mathcal{E}^P} \sum_{\lambda \in \mathcal{D}_{l_1}} b_{vw}^{pqi,l_1,\lambda} \leq \frac{B_{vw}^{\text{REQ}}}{1 - \psi_{\text{HDR}}^{\text{OVH}} \mathbb{1}_{\{\gamma_r=0\}}} \sum_{\sigma \in \mathcal{S}_q} \sum_{k=1}^{K_v^{\text{REQ}}} y_w^{q,\sigma,k}, \quad \forall e_{vw} \in \mathcal{E}_r, q \in \mathcal{V}^{\text{PR}}, w \in \mathcal{V}_r^{\text{R}}, r \in \mathcal{R} \quad (26)$$

where  $l_1 \in \mathcal{L}_{pqi}$  is the link 1 on PP  $e_{pq}^{(i)} \in \mathcal{E}^P$ . According to (24), it is unnecessary to repeat (25) and (26) for all the links.

When revisiting the end nodes of a VP, there will not be a PP needed to connect those nodes. We first define  $h_{vw}^{\text{REV}}$  which is 0 when revisitation occurs for the end nodes of VP  $e_{vw} \in \mathcal{E}_r$ , i.e., nodes are not mapped to the same PS or NR-BS, and 1 otherwise, as follows.

$$\tilde{h}_{vw}^{\text{REV}} \triangleq 1 - \begin{cases} \sum_{p \in \mathcal{V}^{\text{PS}}} \min(x_v^p, x_w^p) & v, w \in \mathcal{V}_r^{\text{S}} \\ \sum_{q \in \mathcal{V}^{\text{PR}}} \min \left( \sum_{\sigma \in \mathcal{S}_q} \sum_{k=1}^{K_v^{\text{REQ}}} y_w^{q,\sigma,k}, \right. \\ \quad \left. \times \sum_{\sigma \in \mathcal{S}_q} \sum_{k=1}^{K_w^{\text{REQ}}} y_w^{q,\sigma,k} \right) & v, w \in \mathcal{V}_r^{\text{R}} \\ 0 & \text{otherwise} \end{cases}$$

Thus, for VP  $e_{vw} \in \mathcal{E}_r$  of SR  $r \in \mathcal{R}$ , we define the effective difference of the required and allocated bandwidth as

$$\hat{h}_{vw}^{\text{B}} \triangleq B_{vw}^{\text{REQ}} h_{vw}^{\text{REV}} - (1 - \psi_{\text{HDR}}^{\text{OVH}} \mathbb{1}_{\{\gamma_r=0\}}) \sum_{e_{pq}^{(i)} \in \mathcal{E}^P} \sum_{\lambda \in \mathcal{D}_{l_1}} b_{vw}^{pqi,l_1,\lambda} \quad (27)$$

To coordinate bandwidth allocation with revisitation,

$$\hat{h}_{vw}^{\text{B}} \geq 0, \quad \forall e_{vw} \in \mathcal{E}_r, r \in \mathcal{R} \quad (28)$$

4) *Maximum Capacity*: For each wavelength slot of each link, the total bandwidth usage must remain within the maximum capacity. Let  $B_{\text{WGB}}^{\text{OVH}}$  denote the overhead of each wavelength guard band at L1. Accordingly, the total utilized bandwidth and bandwidth overhead on wavelength slot  $\lambda \in \mathcal{D}_l$  of link  $l \in \mathcal{L}^P$ , denoted by  $B_{l,\text{TOT}}^{\lambda,\text{UTL}}$  and  $B_{l,\text{TOT}}^{\lambda,\text{OVH}}$ , respectively, can be derived as follows.

$$B_{l,\text{TOT}}^{\lambda,\text{UTL}} = \sum_{r \in \mathcal{R} \cup \mathcal{R}_t^{\text{HP}}} \sum_{e_{vw} \in \mathcal{E}_r} \sum_{e_{pq}^{(i)} \in \mathcal{E}_t^P} \tilde{b}_{vw}^{pqi,l,\lambda} + B^{\text{WSC}} \sum_{r \in \mathcal{R} \cup \mathcal{R}_t^{\text{HP}}} \sum_{e_{vw} \in \mathcal{E}_r} \sum_{e_{pq}^{(i)} \in \mathcal{E}_t^P} \left\lceil \frac{\tilde{b}_{vw}^{pqi,l,\lambda}}{B^{\text{WSC}}} \right\rceil \quad (29)$$

$$B_{l,\text{TOT}}^{\lambda,\text{OVH}} = \sum_{r \in \mathcal{R} \cup \mathcal{R}_t^{\text{HP}}} \sum_{e_{vw} \in \mathcal{E}_r} \sum_{e_{pq}^{(i)} \in \mathcal{E}_t^P} z_{vw}^{l,\lambda} B_{\text{WGB}}^{\text{OVH}} \quad (30)$$

where

$$\tilde{b}_{vw}^{pqi,l,\lambda} \triangleq \begin{cases} b_{vw}^{pqi,l,\lambda}, & \forall e_{vw} \in \mathcal{E}_r, r \in \mathcal{R} \\ B_{vw}^{pqi,l,\lambda}, & \forall e_{vw} \in \mathcal{E}_r, r \in \mathcal{R}_t^{\text{HP}} \end{cases}$$

$$z_{vw}^{l,\lambda} \triangleq \begin{cases} z_{vw}^{l,\lambda}, & \forall e_{vw} \in \mathcal{E}_r, r \in \mathcal{R} \\ Z_{vw}^{l,\lambda}, & \forall e_{vw} \in \mathcal{E}_r, r \in \mathcal{R}_t^{\text{HP}} \end{cases}$$

with  $B_{vw}^{pqi,l,\lambda} \in \mathbb{R}_{\geq 0}$  and  $Z_{vw}^{l,\lambda} \in \{0, 1\}$  denoting the existing VP bandwidth values and mappings for pre-embedded HP SRs, respectively. In addition,  $\mathcal{E}_t^P$ ,  $B^{\text{WSC}} \in \mathbb{R}_{>0}$  and  $B_{\text{WGB}}^{\text{OVH}} \in \mathbb{R}_{>0}$  denotes the set of PPs using link  $l \in \mathcal{L}^P$ , wavelength subcarriers' bandwidth and the overhead of wavelength guard bands, respectively. In (29), for L1 SRs, the ceiling function  $\lceil \cdot \rceil$  is used to round up the bandwidth to the subcarriers' bandwidth and avoid L0 SRs to get access to the possibly remaining bandwidth of those subcarriers. For L0, in the first term of (29), we allocate subcarriers for the cumulative bandwidth of all SRs, as they share subcarriers. We reuse this term for L2 SRs as well.

Therefore, by utilizing (29) and (30), we should enforce

$$B_{l,\text{TOT}}^{\lambda,\text{UTL}} \leq B_{l,\lambda}^{\text{max}} - B_{l,\text{TOT}}^{\lambda,\text{OVH}}, \quad \forall \lambda \in \mathcal{D}_l, l \in \mathcal{L}^P \quad (31)$$

where  $B_{l,\lambda}^{\text{max}}$  denote the maximum bandwidth capacity of link  $l$  on wavelength slot  $\lambda$ .

### E. Profit Constraints

Incorporating isolation levels introduces additional complexity, potentially leading to underutilization of network resources. To mitigate this, in addition to revisitation and migration, one strategy involves permitting controlled QoS violation for LP SRs with a finite-cost penalty in exchange for accepting HP SRs. We need to calculate various costs incurred in the proposed system to derive the profit. These include slice deployment, isolation overheads, migration, and QoS violation costs.

We start by deriving the bandwidth violation cost, denoted by  $g_r^{\text{B}}$ . For SR  $r \in \mathcal{R}$ , we have

$$g_r^{\text{B}} \triangleq \rho_r^{\text{B}} \sum_{e_{vw} \in \mathcal{E}_r} \hat{h}_{vw}^{\text{B}} \quad (32)$$

where  $\rho_r^{\text{B}} \in \mathbb{R}_{\geq 0}$  is the cost of a unit of bandwidth violation. Regarding each VP  $e_{vw} \in \mathcal{E}_r$  of SR  $r \in \mathcal{R}$ 's RTT delay violation cost, we need to consider that delay is determined by the links and not allocated like bandwidth. Therefore, it can either be greater or smaller than the requested delay. Considering this, we derive the effective difference between the requested and experienced delay, denoted by  $\hat{h}_{vw}^{\text{D}}$ , as

$$\hat{h}_{vw}^{\text{D}} \triangleq \left[ \sum_{e_{pq}^{(i)} \in \mathcal{E}^P} \left( D_{pqi}^{\text{RTT}} \min_{l \in \mathcal{L}_{pqi}} \sum_{\lambda \in \mathcal{D}_l} z_{vw}^{l,\lambda} \right) - h_{vw}^{\text{D}} \right]^+ \quad (33)$$

where  $D_{pqi}^{\text{RTT}} \in \mathbb{R}_{>0}$  is the RTT delay of PP  $e_{pq}^{(i)} \in \mathcal{E}^P$  and  $h_{vw}^{\text{D}} \triangleq D_{vw}^{\text{REQ}} + M(1 - \tilde{h}_{vw}^{\text{REV}})$  with  $M \in \mathbb{R}_{>0}$  as a sufficiently large number to cover the case of revisitation.  $[\cdot]^+ \triangleq \max(0, \cdot)$  ensures that only positive delay violations are considered. Thus, we can derive the delay violation cost of SR  $r \in \mathcal{R}$ , denoted by  $g_r^{\text{D}}$ , as

$$g_r^{\text{D}} \triangleq \rho_r^{\text{D}} \sum_{e_{vw} \in \mathcal{E}_r} \hat{h}_{vw}^{\text{D}} \quad (34)$$

Let  $\rho_r^C \in \mathbb{R}_{\geq 0}$  and  $\rho_r^K \in \mathbb{R}_{\geq 0}$  denote the cost of a unit of MIPS and radio resource violation, respectively. We can then derive the MIPS and radio violation costs of  $r \in \mathcal{R}$ , denoted by  $g_r^C$  and  $g_r^K$ , respectively, as

$$g_r^C \triangleq \rho_r^C \sum_{v \in \mathcal{V}_r^S} \left( C_v^{\text{REQ}} - \sum_{p \in \mathcal{V}^{\text{PS}}} c_v^p \right) \quad (35)$$

$$g_r^K \triangleq \rho_r^K \sum_{w \in \mathcal{V}_r^R} \left( K_w^{\text{REQ}} - \sum_{q \in \mathcal{V}^{\text{PR}}} \sum_{\sigma \in \mathcal{S}_q} \sum_{k=1}^{K_w^{\text{REQ}}} k y_w^{q,\sigma,k} \right) \quad (36)$$

Now, by adding (32) and (34)–(36), we can derive SR  $r \in \mathcal{R}$ 's total QoS violation cost, denoted by  $g_r^{\text{VIO}}$ , as

$$g_r^{\text{VIO}} \triangleq g_r^C + g_r^K + g_r^B + g_r^D \quad (37)$$

The next cost is on deployment and the associated overheads. First, let  $\Omega$ ,  $\Phi$ , and  $\Theta$  denote the cost of a unit of MIPS, PRB, and bandwidth. Thus, the deployment cost of L0 and L1 SR  $r \in \mathcal{R}$ , denoted by  $g_r^{\text{DEP,L0}}$  and  $g_r^{\text{DEP,L1}}$ , respectively, is

$$g_r^{\text{DEP,L0}} \triangleq \Phi \sum_{w \in \mathcal{V}_r^R} \sum_{q \in \mathcal{V}^{\text{PR}}} \sum_{\sigma \in \mathcal{S}_q} \sum_{k=1}^{K_w^{\text{REQ}}} k y_w^{q,\sigma,k} + \Omega \sum_{v \in \mathcal{V}_r^S} \sum_{p \in \mathcal{V}^{\text{PS}}} c_v^p$$

$$+ \Theta \sum_{e_{vw} \in \mathcal{E}_r} \sum_{e_{pq}^{(i)} \in \mathcal{E}^{\text{P}}} \sum_{l \in \mathcal{L}_{pq}} \sum_{\lambda \in \mathcal{D}_l} b_{vw}^{pq,i,l,\lambda}$$

$$g_r^{\text{DEP,L1}} \triangleq \Phi N^{\text{FRM}} \sum_{w \in \mathcal{V}_r^R} \sum_{q \in \mathcal{V}^{\text{PR}}} \sum_{\sigma \in \mathcal{S}_q} \sum_{k=1}^{K_w^{\text{REQ}}} \left\lfloor \frac{k y_w^{q,\sigma,k}}{N^{\text{FRM}}} \right\rfloor$$

$$+ \Omega \sum_{v \in \mathcal{V}_r^S} \sum_{p \in \mathcal{V}^{\text{PS}}} c_v^p + \Theta B^{\text{WSC}} \sum_{e_{vw} \in \mathcal{E}_r} \sum_{e_{pq}^{(i)} \in \mathcal{E}^{\text{P}}} \sum_{l \in \mathcal{L}_{pq}} \sum_{\lambda \in \mathcal{D}_l} \left\lfloor \frac{b_{vw}^{pq,i,l,\lambda}}{B^{\text{WSC}}} \right\rfloor$$

For L2 SRs, the deployment cost should consider the allocation of all resources in the mapped PS, NR-BS, and PP. In addition, this should account for when multiple virtual nodes of the same L2 SR are using the same physical entity and avoid overstating the deployment cost. Therefore, the deployment cost, denoted by  $g_r^{\text{DEP,L2}}$ , can be derived as follows.

$$g_r^{\text{DEP,L2}} \triangleq \Phi \sum_{q \in \mathcal{V}^{\text{PR}}} \frac{\prod_{q,0}^{\text{max}}}{\prod_{\text{PRB}}} \max_{\substack{w \in \mathcal{V}_r^R \\ \gamma_r=2}} \sum_{k=1}^{K_w^{\text{REQ}}} y_w^{q,0,k} \\ + \Omega \sum_{p \in \mathcal{V}^{\text{PS}}} C_p^{\text{max}} \max_{\substack{v \in \mathcal{V}_r^S \\ \gamma_r=2}} x_v^p + \Theta \sum_{l \in \mathcal{L}} B_{l,0}^{\text{max}} \max_{\substack{e_{vw} \in \mathcal{E}_r \\ \gamma_r=2}} z_{vw}^{l,0}$$

According to Fig. 2, overheads are not necessarily specific to individual SRs, e.g., hypervisor or host OS. Therefore, we derive the total overhead costs, denoted by  $g_{\text{TOT}}^{\text{OVH}}$ , as follows.

$$g_{\text{TOT}}^{\text{OVH}} \triangleq \Phi \sum_{q \in \mathcal{V}^{\text{PR}}} \sum_{\sigma \in \mathcal{S}_q} \Pi_{q,\text{TOT}}^{\sigma,\text{OVH}} + \Omega \sum_{p \in \mathcal{V}^{\text{PS}}} C_{p,\text{TOT}}^{\text{OVH}} \\ + \Theta \sum_{l \in \mathcal{L}^{\text{P}}} \sum_{\lambda \in \mathcal{D}_l} B_{l,\text{TOT}}^{\lambda,\text{OVH}}$$

Migration of RU and NF nodes of pre-embedded slices must be associated with a cost, denoted by  $g_{\hat{r}}^{\text{MIG}}$  for SR  $\hat{r} \in \mathcal{R}_t$ , that depends on the slice priority. Thus,

$$g_{\hat{r}}^{\text{MIG}} \triangleq \beta_{\hat{r}} \sum_{v \in \mathcal{V}_{\hat{r}}^S} \sum_{p \in \mathcal{V}^{\text{PS}}} x_v^p (1 - X_v^p) \\ + \beta_{\hat{r}} \sum_{w \in \mathcal{V}_{\hat{r}}^R} \sum_{q \in \mathcal{V}^{\text{PR}}} \left( \sum_{\sigma \in \mathcal{S}_q} \sum_{k=1}^{K_w^{\text{REQ}}} y_w^{q,\sigma,k} \right) \left( 1 - \sum_{\sigma \in \mathcal{S}_q} \sum_{k=1}^{K_w^{\text{REQ}}} Y_w^{q,\sigma,k} \right)$$

where  $Y_w^{q,\sigma,k} \in \{0, 1\}$  denote the existing mapping decisions of RU  $w \in \mathcal{V}_{\hat{r}}^R$  of SR  $\hat{r} \in \mathcal{R}_t$ , respectively.

The InP wants to ensure that the added profit after admitting  $r_t$  at time step  $t$  is higher than a minimum  $P^{\text{min}}$ . With  $g_r^{\text{DEP}} \triangleq g_r^{\text{DEP,L0}} + g_r^{\text{DEP,L1}} + g_r^{\text{DEP,L2}}$ , we first define the InP's obtained profit at time step  $t$  as follows.

$$P_t \triangleq \sum_{r \in \mathcal{R} \cup \mathcal{R}_t^{\text{HP}}} R_r - \sum_{r \in \mathcal{R}} (g_r^{\text{DEP}} - g_r^{\text{VIO}}) - \sum_{\hat{r} \in \mathcal{R}_t} g_{\hat{r}}^{\text{MIG}} - g_{\text{TOT}}^{\text{OVH}} \\ - g_{\text{HP},t}^{\text{DEP}}$$

where  $g_{\text{HP},t}^{\text{DEP}} \in \mathbb{R}_{\geq 0}$  is the total deployment cost of pre-embedded HP SRs at time step  $t$ .

$$P_t - P_{t-1} \geq P^{\text{min}} \quad (38)$$

where  $P_{t-1} \in \mathbb{R}_{\geq 0}$  is given from time step  $t-1$ .

## F. 5G Isolation-Aware Network Slicing (5G-INS) Problem

We consider the objective to be the cumulative profit the InP expects to generate if it admits  $r_t$ . Therefore, we can define the 5G-INS problem as follows.

$$\text{maximize}_{\mathbf{x}, \mathbf{y}, \mathbf{z}, \mathbf{c}, \mathbf{b}} P_t \quad (39a)$$

$$\text{subject to (1), (2), (4) – (9), (15), (16), (19) – (26), (39b)}$$

$$(28), (31), (38) \quad (39c)$$

## G. Analysis and Simplification

5G-INS is a mixed-integer programming (MIP) problem that contains non-linear, discontinuous functions such as the indicator function. This alone does not mean that 5G-INS is intractable. However, the following theorem applies.

*Theorem 1.* 5G-INS is an NP-hard problem.

*Proof.* Refer to Appendix A of the supplemental material, available online.  $\square$

Therefore, 5G-INS is intractable, and we cannot trivially tackle 5G-INS due to NP-hardness. However, we present the Theorem below, which makes the reduction of 5G-INS to a tractable form possible.

*Theorem 2.* We reduce 5G-INS into a linear program by relaxing the binary input variables to the range  $[0, 1]$ .

*Proof.* Refer to Appendix B of the supplemental material, available online.  $\square$

Another aspect of 5G-INS to analyze is its integrality gap, i.e., the maximum ratio of the original problem's optimal solution to the relaxed problem's, which can reveal how rounding the binary

**Algorithm 1:** 5Guard Framework.

---

**Input:**  $\mathcal{V}^{PS}, \mathcal{V}^{PR}, \mathcal{E}^P, \mathcal{L}^P, \mathcal{R}_t, r_t, P^{\min}, \Pi^{PRB}, N^{FRM}, t$

- 1  $\mathcal{A} \triangleq \{A_i | A_i \text{ is an optimization algorithm}\};$
- 2  $T^{STA} = \text{time}()$
- 3 Instantiate  $|\mathcal{A}|$  number of processing threads
- 4 **cobegin**  $A_i$
- 5   Execute  $A_i$  on the input to solve 5G-INS at time step  $t$
- 6 **while** All  $A_i$ s are not finished AND  $\text{time}() - T^{STA} < T_{r_t}$  **do**
- 7   continue
- 8  $\mathcal{A}^{RET} \triangleq \{A_i | A_i \text{ returned a feasible solution}\}$
- 9 **if**  $\mathcal{A}^{RET} = \emptyset$  **then**
- 10   **return** *Infeasible*
- 11  $i^* = \text{argmax}(\text{profit}(\mathcal{A}^{RET}))$
- 12  $\mathbf{x}^*, \mathbf{y}^*, \mathbf{z}^*, \mathbf{c}^*, \mathbf{b}^* = \text{argmax}(A_{i^*}^{RET})$

**Output:**  $\mathbf{x}^*, \mathbf{y}^*, \mathbf{z}^*, \mathbf{c}^*, \mathbf{b}^*$

---

variables affects the optimal solution. We present the following Theorem.

*Theorem 3.* The integrality gap of 5G-INS is unbounded.

*Proof.* Refer to Appendix C of the supplemental material, available online.  $\square$

In the following section, we will leverage Theorems 2 and 3 to propose a framework that effectively addresses the complexity of 5G-INS.

## V. 5GUARD FRAMEWORK

In this section, we propose 5Guard, a framework to address 5G-INS. We then introduce four algorithms with distinct characteristics for use within 5Guard and analyze the complexity.

### A. Proposed Framework and Algorithms

According to Theorem 2, as 5G-INS can be reduced to an LP for relaxed binary variables, a range of algorithms can be chosen. While state-of-the-art algorithms sometimes offer guarantees for worst-case scenarios, this does not ensure superior performance in all situations. Specifically, algorithms with better worst-case guarantees may not always outperform others in average cases, where performance can vary based on the context. Moreover, modern integer LP (ILP) solvers, such as Gurobi, leverage advanced techniques like branch-and-cut combined with the barrier method [38] to exploit multiple cores, enhancing computational efficiency while guaranteeing optimality. This implies that optimal solutions may be achievable for slices with lenient resource allocation deadlines. Thus, individual algorithms have tradeoffs in execution time or optimality.

As illustrated in Fig. 3, to address this issue, we propose 5Guard, a framework that leverages an ensemble of optimization algorithms and exploits parallel processing to identify the best solution within the resource allocation deadline. The pseudo-code of 5Guard for each time step  $t$  is provided in Algorithm 1. Here, time steps represent the iterations of 5Guard, which is distinct from execution time. In Line 1, we define the set  $\mathcal{A}$  as the ensemble of optimization algorithms. We first measure the start time in Line 2 with the function  $\text{time}()$ . In Line 3,

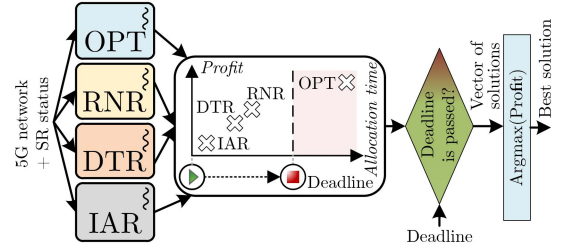


Fig. 3. Flow diagram of the 5Guard framework for a 5G-INS instance. The ensemble algorithms, e.g., IAR, DTR, RNR, and OPT, operate concurrently on separate threads, processing shared input data. If the resource allocation deadline is reached, all algorithm executions are halted, and the best feasible solution is selected.

we instantiate  $|\mathcal{A}|$  processing threads to run the algorithms in parallel. In Lines 4 and 5, “cobegin  $A_i$ ” indicates the parallel block of code parameterized with algorithm  $A_i$ . In Lines 6 and 7, 5Guard waits until all the algorithms complete execution or the elapsed time exceeds the resource allocation deadline  $T_{r_t}$  of SR  $r_t$ . In Line 8, we store the algorithms that return a feasible solution in set  $\mathcal{A}^{RET}$ . If  $\mathcal{A}^{RET}$  is empty, it implies that no algorithm could reach a feasible solution or finish in time. Thus, 5Guard returns infeasible, hence rejecting  $r_t$ . Otherwise, it chooses the index  $i^*$  that has achieved the maximum profit. It finally returns  $A_{i^*}^{RET}$ 's decisions.

For this work, as a proof-of-concept, we design set  $\mathcal{A}$  from four custom algorithms: 1) Isolation-aware ranking (IAR), 2) deterministic rounding (DTR), 3) randomized rounding (RNR), and 4) exact (OPT), where OPT uses Gurobi's solver based on branch-and-cut and barrier method. These algorithms present a wide range of complexity and approximation ratios. Based on the unbounded integrality gap of 5G-INS proven in Theorem 3, only relying on rounding techniques can be limiting.

The IAR algorithm is outlined in Algorithm 2. We first define  $\mathbf{x}^*$ ,  $\mathbf{y}^*$ , and  $\mathbf{z}^*$  as placeholder vectors for optimal values of  $\mathbf{x}$ ,  $\mathbf{y}$ , and  $\mathbf{z}$ . Next, we keep the mappings of the pre-embedded HP SRs in Line 2 based on time step  $t - 1$ . In Line 5, we iteratively rank the remaining SRs, defined as  $\mathcal{R}^M$ , by profit, isolation level, and priority. In Line 6 and 7, considering the complete isolation required by L2, we keep track of the remaining RUs of selected SR  $r^*$  and NR-BSSs, defined as  $\mathcal{V}_{r^*}^{R,M}$  and  $\mathcal{V}^{PR,M}$ , respectively. In Line 10, to avoid bottlenecks, we select RUs and non-isolated NR-BSSs with the highest radio resource demand and remaining radio resources, respectively. In addition, the selected RU receives the maximum possible PRBs. Accordingly, the selected RU, NR-BS, frequency slot, and number of PRBs, denoted by  $w^*$ ,  $q^*$ ,  $f^*$ ,  $k^*$ , respectively, will be obtained as follows.

$$\begin{aligned}
 w^* &= \text{argmax}_{w \in \mathcal{V}_{r^*}^{R,M}} (K_w^{\text{REQ}}) \\
 q^* &= \text{argmax}_{q \in \mathcal{V}^{PR,M}} (\max_{f \in \mathcal{F}_q} \Pi_{q,f}^{\text{REM}}) \\
 f^* &= \text{argmax}_{f \in \mathcal{F}_{q^*}} (\Pi_{q^*,f}^{\text{REM}}) \\
 k^* &= \min(K_{w^*}^{\text{REQ}}, \Pi_{q^*,f^*}^{\text{REM}})
 \end{aligned} \tag{40}$$

**Algorithm 2:** 5Guard Framework.

---

**Input:**  $\mathcal{V}^{\text{PS}}, \mathcal{V}^{\text{PR}}, \mathcal{E}^{\text{P}}, \mathcal{L}^{\text{P}}, \mathcal{R}_t, r_t, P^{\text{min}}, \Pi^{\text{PRB}}, N^{\text{FRM}}, t$

- 1  $\mathbf{x}^* = \mathbf{y}^* = \mathbf{z}^* = \mathbf{0}$
- 2 Maintain the mappings for pre-embedded HP SRs from  $t - 1$
- 3  $\mathcal{R}^{\text{M}} = \mathcal{R}$
- 4 **while**  $\mathcal{R}^{\text{M}} \neq \emptyset$  **do**
- 5      $r^* = \operatorname{argmax}_{r \in \mathcal{R}^{\text{M}}} R_r(\gamma_r + 1)(\beta_r + 1)$
- 6      $\mathcal{V}_{r^*}^{\text{R,M}} = \mathcal{V}_{r^*}^{\text{R}}$
- 7      $\mathcal{V}_{r^*}^{\text{P,R,M}} = \mathcal{V}^{\text{P,R}} \setminus \{\text{Completely isolated NR-BSs}\}$
- 8     **while**  $\mathcal{V}_{r^*}^{\text{P,R,M}} \neq \emptyset$  **do**
- 9         **if**  $\mathcal{V}_{r^*}^{\text{P,R,M}} = \emptyset$  **then return infeasible**
- 10         Set  $w^*, q^*, f^*$ , and  $k^*$  based on (40)
- 11         **if**  $k^* = 0$  **then return infeasible**
- 12          $(y_{w^*}^{q^*, f^*, k^*})^* = 1$
- 13         **if**  $\gamma_{r^*} = 2$  **then** Mark  $q^*$  as completely isolated
- 14          $\mathcal{V}_{r^*}^{\text{R,M}} = \mathcal{V}_{r^*}^{\text{R,M}} \setminus \{w^*\}$
- 15     Repeat Lines 6-12 for PSs to obtain  $\mathbf{x}^*$
- 16     **while**  $\mathcal{E}_{r^*}^{\text{M}} \neq \emptyset$  **do**
- 17          $e_{vw}^* = \operatorname{argmax}_{e_{vw} \in \mathcal{E}_{r^*}^{\text{M}}} B_{vw}^{\text{REQ}}$
- 18          $\mathcal{E}^{\text{P,M}}$  = set of PPs connecting mapped  $v$  and  $w$
- 19         **if**  $\tilde{h}_{vw}^{\text{REV}} = 1$  **then**
- 20              $e_{pq}^{(i)*} = \operatorname{argmax}_{e_{pq}^{(i)} \in \mathcal{E}^{\text{P,M}}} (\min_{l \in \mathcal{L}_{pq}, \lambda \in \mathcal{D}_l} B_{l,\lambda}^{\text{REM}})$
- 21             **for**  $l^* \in \mathcal{L}_{(pq)^*}$  **do**
- 22                  $\lambda^* = \operatorname{argmax}_{\lambda \in \mathcal{D}_{l^*}} (B_{l^*,\lambda}^{\text{AVA}})$
- 23                  $(z_{(vw)^*}^{l^*, s^*})^* = 1$
- 24              $\mathcal{E}_{r^*}^{\text{M}} = \mathcal{E}_{r^*}^{\text{M}} \setminus \{e_{vw}^*\}$
- 25      $\mathcal{R}^{\text{M}} = \mathcal{R}^{\text{M}} \setminus \{r^*\}$
- 26 Solve 5G-INS for  $(\mathbf{x}, \mathbf{y}, \mathbf{z}) = (\mathbf{x}^*, \mathbf{y}^*, \mathbf{z}^*)$  and obtain  $\mathbf{c}^*, \mathbf{b}^*$

**Output:**  $\mathbf{x}^*, \mathbf{y}^*, \mathbf{z}^*, \mathbf{c}^*, \mathbf{b}^*$

---

where

$$\Pi_{q,f}^{\text{REM}} \triangleq \Pi_{q,f}^{\text{max}} - \sum_{r \in \mathcal{R}} \sum_{w \in \mathcal{V}_r^{\text{R}}} \sum_{q \in \mathcal{V}^{\text{P,R}}} \sum_{f \in \mathcal{F}_q} (y_w^{q,f,k})^*$$

denotes the remaining radio resource on frequency slot  $f$  of NR-BS  $q$  at each iteration.

If L2 isolation is required, we mark the selected NR-BS from the remaining set  $\mathcal{V}^{\text{P,R,M}}$  in Line 13 to be excluded from the next iteration in Line 7. We similarly repeat these steps for PSs in Line 15, assuming a fully connected 5G network. We select VPs based on bandwidth demand, similar to RUs and NFs in Line 17. However, we first have to check whether revisitation has occurred for the end nodes of the selected VP. The revisitation of end nodes results in skipping the VP. Otherwise, we choose greedily the PP, links, and wavelength slots as outlined in Lines 20–23. In case of infeasibility, the algorithm rejects  $r_t$  as pre-admitted slices cannot be dismissed. Otherwise, IAR solves the 5G-INS problem with the obtained mappings and outputs the solution.

Finally, for rounding algorithms, i.e., DTR and RNR, we first use Theorem 2 to solve 5G-INS with relaxed binary variables in polynomial time. For deterministic rounding, we utilize a fixed limit to round the potentially fractional values for binary mapping variables, keeping in mind that rounded values should not violate any constraint. For randomized rounding, we treat the solution of the relaxed problem for binary variables as

probabilities for rounding, which again should adhere to the constraints.

### B. Complexity Analysis

Since 5Guard is a parallel algorithm, we should analyze both *depth* and *work* for its complexity [39]. *Depth* is defined as the time complexity of the longest series of computations executed sequentially. *Work* denotes the total number of primitive works executed by the parallel processor. As the allocation time is limited to  $T_{r_t}$  for SR  $r_t$ , depth is of the order  $\mathcal{O}(T_{r_t})$ . If  $T_{r_t}$  is unrestricted, the depth is limited by the slowest performing algorithm. If  $\mathcal{A}$  includes an exact algorithm such as branch-and-cut, 5Guard will continue executing until the optimal solution is found. Let  $T^{\text{max}}$  denote the highest time complexity among the algorithms in set  $\mathcal{A}$ . In the worst case, all the algorithms in  $\mathcal{A}$  finish their execution. Therefore, the depth complexity will be of  $\mathcal{O}(T^{\text{max}})$ , and work complexity will be of  $\mathcal{O}(\sum_{A_i \in \mathcal{A}} T^{\text{max}}) = \mathcal{O}(T^{\text{max}})$ . This means that having lower complexity algorithms to support challenging requirements does not significantly impact the overall processing work done by the decision-making processor. Although this can be of exponential order depending on the algorithms in  $\mathcal{A}$ , we must note that the depth and work complexity are fully controllable by  $T_{r_t}$  to achieve the best solution in time.

When 5Guard contains DTR, RNR, and IAR as heuristics in the ensemble, their complexities can be analyzed as follows. Both DTR and RNR involve a rounding phase with worst-case complexity  $\mathcal{O}(L \cdot S \cdot E)$ , proportional to the number of binary variables in the link mapping  $\mathbf{z}$ , where  $L \triangleq |\mathcal{L}^{\text{P}}|$ ,  $S \triangleq \max_{l \in \mathcal{L}^{\text{P}}} \mathcal{D}_l$ ,  $E \triangleq \max_{r \in \mathcal{R}} \mathcal{E}_r$ . For IAR, the mapping process (i.e., Lines 4–25 in Algorithm 2) includes three phases: 1) NR-BS mapping, 2) PS mapping, and 3) VP mapping. Assuming linear-time “argmax” operations and defining  $R \triangleq |\mathcal{R}|$ ,  $V_R \triangleq |\mathcal{V}^{\text{P,R}}|$ , and  $V_S \triangleq |\mathcal{V}^{\text{P,S}}|$ , the complexities are:  $\mathcal{O}(V_R^2)$  for NR-BS mapping,  $\mathcal{O}(V_S^2)$  for PS mapping, and  $\mathcal{O}(E \cdot E_P \cdot L \cdot S)$  for VP mapping. Considering the outer while loop iterates over all  $R$  SRs, the total complexity is  $\mathcal{O}(R \cdot (V_R^2 + V_S^2 + E \cdot E_P \cdot L \cdot S))$ . Nonetheless, the dominant time cost arises from solving the optimization problems rather than these steps.

The three algorithms discussed above rely on solving LP forms of the 5G-INS problem. This paper uses barrier methods, a subset of interior-point methods characterized by a logarithmic barrier function, to solve the problem. This method exhibits a computational time complexity of  $\mathcal{O}(M^{1.5} \cdot N^2)$  [40], where  $M$  and  $N$  denote the number of constraints and variables, respectively. This complexity also depends on parameters controlling input and solution precision. According to constraint (23) and considering that the 5G substrate network is often a complete graph, i.e., number of edges larger than the nodes,  $M$  and  $N$  will both be in the order of  $\mathcal{O}(L \cdot S \cdot E \cdot R)$ . This means that these three algorithms are of polynomial-time complexity, making them scalable with the size of the substrate network and SR load. Note that the reductions made in Theorem 2 and the max operations in (40) do not change the time complexity. Although the order of complexity is the same for these algorithms, the isolation-aware ranking algorithm proves to be the fastest in

TABLE IV  
KEY PARAMETERS SETTING

Par.	Value	Par.	Value	Par.	Value
$\Pi^{\text{PRB}}$	180	$\Pi_{\text{FGB}}^{\text{OVH}}$	693	$N^{\text{FRM}}$	10
$\epsilon$	1Hz	$\Pi_{q,0}^{\text{max}}$	500,040	$\psi_{\text{HDR}}^{\text{OVH}}$	0.005
$M$	1sec.	$B^{\text{WSC}}, B^{\text{WGB}}$	12.5 GHz	$B_{l,0}^{\text{max}}$	4.8 THz
$C_{\text{OVH}}^{\text{GOS}}$	25,000	$C_p^{\text{max}}$	834,797 MIPS,	$\delta$	$10^{-6}$
$C_{\text{OVH}}^{\text{K8S}}$	MIPS	(MEC, Cloud)	8,347,970 MIPS		MIPS
$\chi_S^{\text{max}}$	6	$C_{\text{OVH}}^{\text{HHO}}$	100,000 MIPS		

general, due to having considerably fewer constraints and variables (i.e.,  $\mathbf{b}$  and  $\mathbf{c}$ ) in the relaxed problem. This makes it suitable for SRs requiring a fast resource allocation time, such as MC services.

## VI. EVALUATION

In this section, we evaluate the performance of 5Guard<sup>1</sup> by analyzing various metrics across different isolation levels. We also examine the overheads of integrating isolation into 5G NS across different resource domains. Finally, we discuss the key remarks and limitations of this study.

### A. Simulation Setup

For the 5G system, we follow the values used in the specifications [36], [37], [41], as listed in Table IV. These values are derived based on the following assumptions. PRBs have 12 subcarriers, 15 KHz spacing, and 1 ms interval [36]. Radio frame intervals are set to 10 ms [36], and NR-BS bandwidth to 50 MHz [36]. MEC servers use AMD EPYC 9684X, and 10 such servers are deployed in the cloud. Based on their benchmark results, each server outputs 834,797 MIPS [42]. The wavelength range is 191.3 THz-196.1 THz as used in the industry [37]. The MPLS protocol employs 32-bit headers [41] and a maximum IPv4 packet size of 65535 bits. Our resource allocator testbed utilizes a server-grade Intel Xeon Silver 4314 CPU with 64 threads and 1 TB of RAM to simulate a 5G InP setup.

In our simulation, we evaluate 5Guard's performance and analyze the impact of isolation on 5G NS in resource-critical scenarios and large-scale topologies. For the first goal, as shown in Fig. 4(a), we design a synthetic topology resembling a small MEC/telco cloud network, fully connected to NR-BSs, with a small amount of resources. For large-scale evaluation, as shown in Fig. 4(b), we use a scaled version of the Berlin Research Area Information Network (BRAIN) [43] as a real-world network topology for our 5G MEC/telco cloud network, featuring 51 PSs, 48 NR-BSs, and 152 PPs. The BRAIN's 5G topology follows the resource values in Table III, while the synthetic topology assumes only 20% of these resources to model a more resource-critical scenario.

We consider four types of slices: an L2 MC slice, an L1 uRLLC slice, an L0 mMTC slice without VNF sharing, and an L0 eMBB streaming slice with VNF sharing. To model a more

<sup>1</sup>5Guard's repository is available at [github.com/mehdibolourian/5Guard](https://github.com/mehdibolourian/5Guard), allowing for result reproduction and experiment extensions.

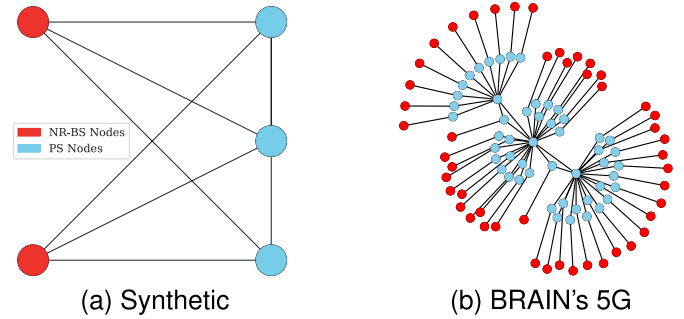


Fig. 4. 5G infrastructure topologies used in the evaluation.

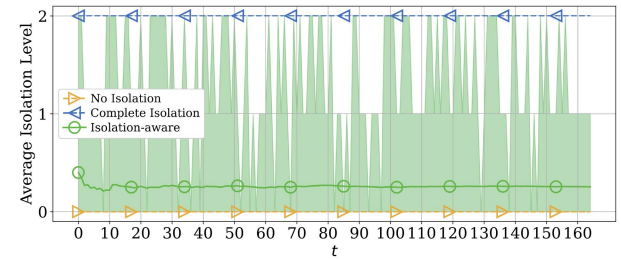


Fig. 5. The average SR's isolation level fed to isolation-aware approaches, the approach with no isolation, and complete isolation at each time step, with the shaded area showing the range of absolute isolation levels across the iterations.

realistic arrival process, SRs of these slices arrive randomly at each time step with probabilities of 0.7, 0.15, 0.1, and 0.05 for eMBB, uRLLC, mMTC, and MC slices, respectively. The results are averaged over multiple runs.

Fig. 5 shows the average isolation levels selected for arriving SRs at each time step  $t$ , with the shaded area indicating the range of selected isolation levels. Regarding resource allocation deadlines, since provisioning resources (e.g., K8s pods or VMs) typically takes at least seconds, deadlines should remain within the same order to avoid altering service access times. For our simulation, we set deadlines to 2sec. for MC slices, 4sec. for mMTC slices, 6sec. for uRLLC slices, and 10sec. for eMBB slices.

Inspired by [44], we adopt a pay-as-you-go revenue collection scheme, commonly used by InPs for service pricing. In our approach, pricing is based on deployment costs, which are significantly influenced by the isolation level imposed on the InP. These costs are then scaled by a coefficient that adjusts the profit, considering other factors, e.g., maintenance expenses.

### B. Simulation Results

1) *5Guard Evaluation*: Throughout the simulation runs, we modeled SR arrivals at each time step  $t$ . Fig. 6(a)–(d) present the SR admission percentages and profit, alongside the cumulative number of migrations and violation costs using the synthetic topology shown in Fig. 4(a). Migration and violation costs correspond to the two previously introduced techniques to increase admission.

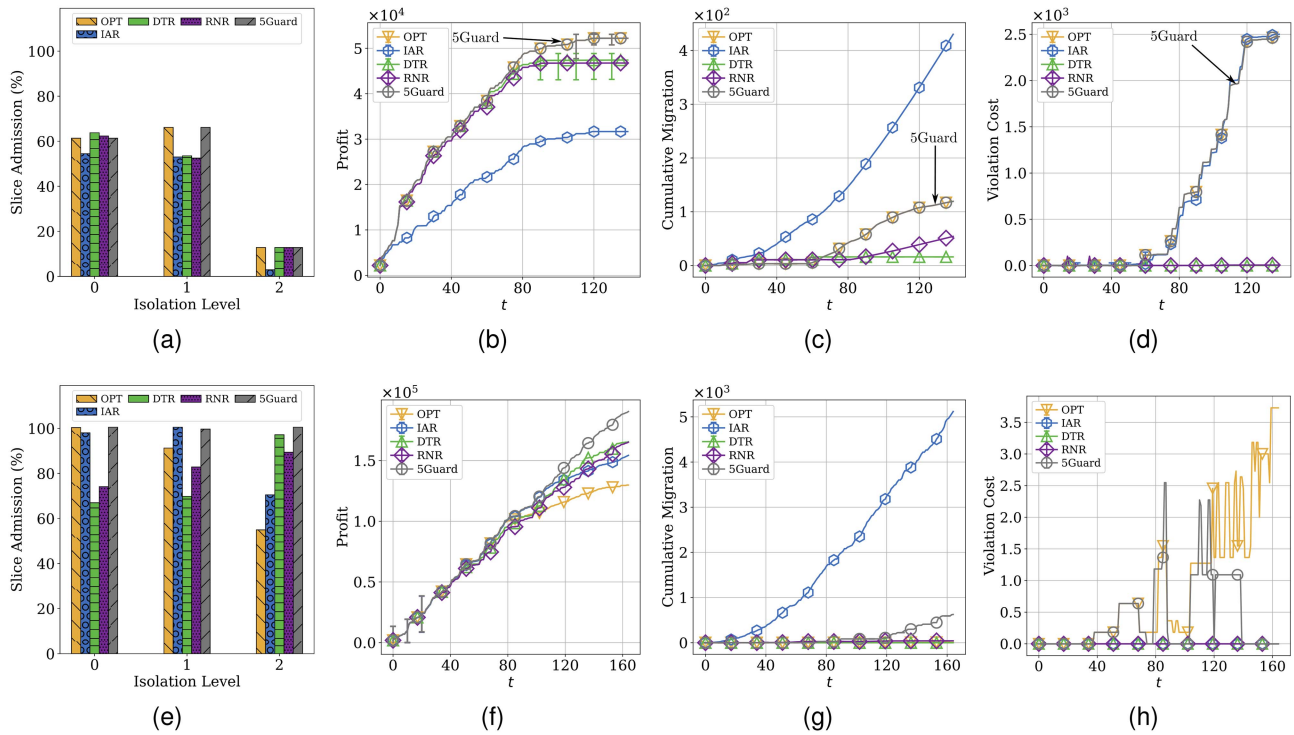


Fig. 6. Evaluation of SR admission percentage for each isolation level, profit, cumulative number of migrations, and violation cost over time steps by incrementally adding a new SR with different isolation levels at each time step for small-scale (i.e., a–d) and large-scale (i.e., e–h) 5G topologies.

As mentioned earlier, 5G-INS is the first isolation-aware formulation of 5G NS that integrates isolation levels according to standards. Consequently, comparing 5Guard with existing baselines that lack support for these levels may lead to an unfair evaluation due to differing SR inputs and, hence, different revenues under the pay-as-you-go scheme. According to Theorem 2, we proved that 5G-INS can be accurately transformed into a linear program with relaxed binary variables. Thus, OPT represents the optimal solution to 5G-INS, assuming all SRs are admitted whenever constraints allow it.

As shown in Fig. 6(a)–(d), 5Guard matches the performance of OPT. This is because the topology is small, allowing OPT to meet the deadlines for all SRs, including the L2 MC service for incoming SR  $r_t \in \mathcal{R}$  at each time step  $t$ . Consequently, 5Guard achieved up to 10.4% higher profit than DTR, the best-performing individual algorithm, and matched OPT. Since this topology includes only two NR-BS nodes, accepting more than one additional L2 SR is impossible after admitting one SR, as rejecting SRs post-admission is not allowed.

As depicted in Fig. 6(a), 5Guard successfully admitted an L2 SR, achieving a non-zero admission rate, while allocating the remaining NR-BS/PS resources to L0/L1 SRs. Fig. 6(c), (d), (g), and (h) illustrate how QoS violation and service migration strategies maximize profit. As stated earlier, the synthetic topology is designed to simulate a resource-constrained scenario. This demonstrates that controlled violation costs can significantly increase the admission of LP SRs in such conditions, where admission would otherwise be impossible.

In Fig. 6(e)–(h), we extended our investigation to the large-scale BRAIN’s 5G topology shown in Fig. 4(b). 5Guard achieved

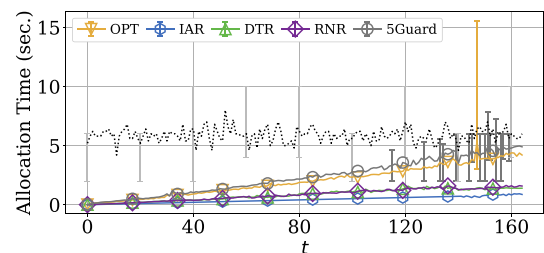


Fig. 7. Resource allocation times with the BRAIN’s 5G topology at each time step, with the dashed line indicating the average resource allocation deadlines of SRs alongside the error bars showing the range.

up to 15.1% more average profit than the best-performing algorithm in the ensemble and up to 46.7% more average profit than OPT. It also achieved up to 33.5% more admissions compared to the best-performing algorithm. The significant increase in input parameters caused a notable rise in OPT’s execution time, occasionally leading to failures in meeting the resource allocation deadlines for MC and uRLLC SRs. However, we observed that running OPT for L0 SRs helped guide the allocation to superior solutions for later time steps, positively influencing the results even when switching to other algorithms. Without loss of generality, we assumed resource allocation deadlines of 2 sec., 4 sec., 6 sec., and 10 sec. for MC, uRLLC, eMBB, and mMTC SRs, respectively. Note that these values differ from the delay requirements of SRs, which can be on the order of microseconds.

According to Fig. 7, from  $t = 70$ , OPT failed to meet the MC SR requirements, and for  $t > 128$ , it also could not meet the uRLLC SR requirements. Additionally, error bars of 5Guard’s

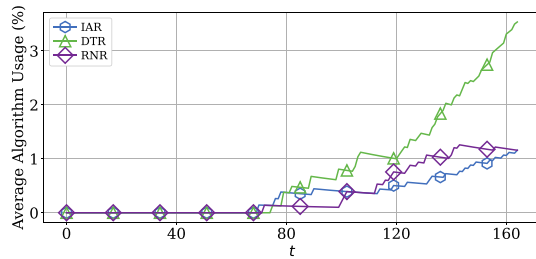


Fig. 8. Running average percentage of usage for each non-OPT algorithm in 5Guard's ensemble while running on the BRAIN's 5G topology.

allocation time for  $t > 70$  demonstrate ranges with minimum and maximum values set to 2sec. and 4sec., respectively. This behavior is due to 5Guard continuing the execution of algorithms up to the SR's deadline, as outlined in Algorithm 1. Furthermore, while OPT operates in a large-scale topology with many SRs, its allocation time does not show a drastic increase compared to other algorithms in the ensemble. However, based on our observations during the simulations, OPT can exhibit unreliability in specific scenarios, resulting in considerably larger allocation times. This is evidenced by the increasing allocation times and error bar spike in Fig. 7, where OPT's runtime reaches over 15sec. for some instances at  $t = 148$ . This does not suggest that OPT is consistently unreliable or slow, primarily due to the use of advanced techniques such as branch-and-cut, especially in testbeds with many CPU cores allocated to the algorithms.

Fig. 8 illustrates the running average of usage percentages for non-OPT algorithms within the 5Guard framework. The results indicate that DTR is the most frequently selected algorithm, followed by IAR and RNR. Note that the percentage is controlled by how many L1 and L2 SRs we feed to the system, which have more stringent deadlines. This experiment helps determine which algorithms to include in the ensemble to minimize algorithm overhead while maximizing performance. As depicted in Fig. 6(f), algorithms within the ensemble continuously surpass each other in profit, stemming from their distinct optimality-complexity trade-offs discussed earlier. As a result, all algorithms have been utilized by 5Guard, achieving 100% admissions across the isolation levels.

Executing multiple algorithms concurrently can introduce additional overhead for the 5G InP. Based on our findings, in scenarios where the OPT algorithm consistently meets deadlines, the ensemble can be streamlined to include only OPT, thereby reducing computational complexity. Conversely, in environments characterized by expansive network topologies and stringent deadlines, excluding OPT from the ensemble may be advantageous due to its computational demands. In other situations, performance may vary over time, with different algorithms outperforming others depending on specific conditions. Therefore, incorporating diverse algorithms within the ensemble can enhance online performance by leveraging their complementary strengths. Developing adaptive strategies for ensemble management, which dynamically adjust the algorithm set based on real-time network conditions, presents a promising direction for future research.

Fig. 6(g) demonstrates that IAR caused significantly more migrations than the other algorithms. This is because it selects NR-BSs, PSs, and PPs with the maximum remaining resources such that when a new SR with higher revenue, isolation level, and priority than one or more pre-embedded SRs arrives, it can trigger the migration of all SRs with lower values.

2) *Isolation Overhead Analysis*: In Fig. 9(a)–(h), we present the overhead costs for different algorithms, with the two rows corresponding to the synthetic and BRAIN topologies, respectively. Additionally, we introduce two new baselines representing extreme cases of isolation-aware NS: NIS, which applies no isolation for all SRs, and CIS, which enforces complete isolation for all SRs. To compute overheads, we consider the resource portions highlighted in red in Fig. 2.

A key takeaway from these results is that, under NIS, the 5G InP experiences no overhead on radio resources and minimal overhead on bandwidth, but significantly higher overhead on MIPS resources. This is due to the absence of additional isolation guard bands for bandwidth and radio resources at L0. However, proper operation at L0 still requires extra headers for data transmission on VPs and additional infrastructure, such as OpenStack, a hypervisor, host and guest OSs, and a container orchestration tool for NFs.

Another observation is that overhead generally increases with the number of admitted SRs. However, MIPS overhead is also influenced by the placement of these SRs. OPT and 5Guard, which admit more SRs, also incur higher overhead costs. Notably, MIPS overhead spikes whenever a new VM or PS is used for the first time, which can significantly impact the 5G InP's profit at those time steps. Therefore, the InP should carefully set  $P_{\min}$  in (38) to account for these fluctuations.

Additionally, IAR demonstrates significantly higher MIPS overhead in the BRAIN's 5G topology. This stems from IAR's strategy of selecting NR-BS and PS nodes with the highest remaining resources, allocating a new, previously unused PS at each time step until all PSs are assigned at least one SR. The results show that the MIPS overhead has the steepest increase up to  $t = 51$ , corresponding to the number of PSs in the BRAIN topology.

Another insight is that radio overhead cost is the primary contributor to total overhead cost. This is particularly evident when the NR-BS bandwidth is 50 MHz, translating to approximately 2778 PRBs. Given the resource requirements, this can become a bottleneck, leading to a higher cost per resource unit and, consequently, a higher overall overhead cost.

The other aspect of isolation we study is its impact on resource usage patterns, as discussed earlier. Fig. 9(i)–(p) illustrate the average and domain-specific resource efficiency for the synthetic and the BRAIN's 5G topologies. These figures account for the overhead costs and the resource usage patterns mentioned previously.

As expected, enforcing complete isolation results in the lowest possible resource efficiency, whereas the absence of isolation yields the highest efficiency. For MIPS resources, efficiency starts at a low value and gradually increases as more SRs are admitted. This is because, initially, a significant portion of the

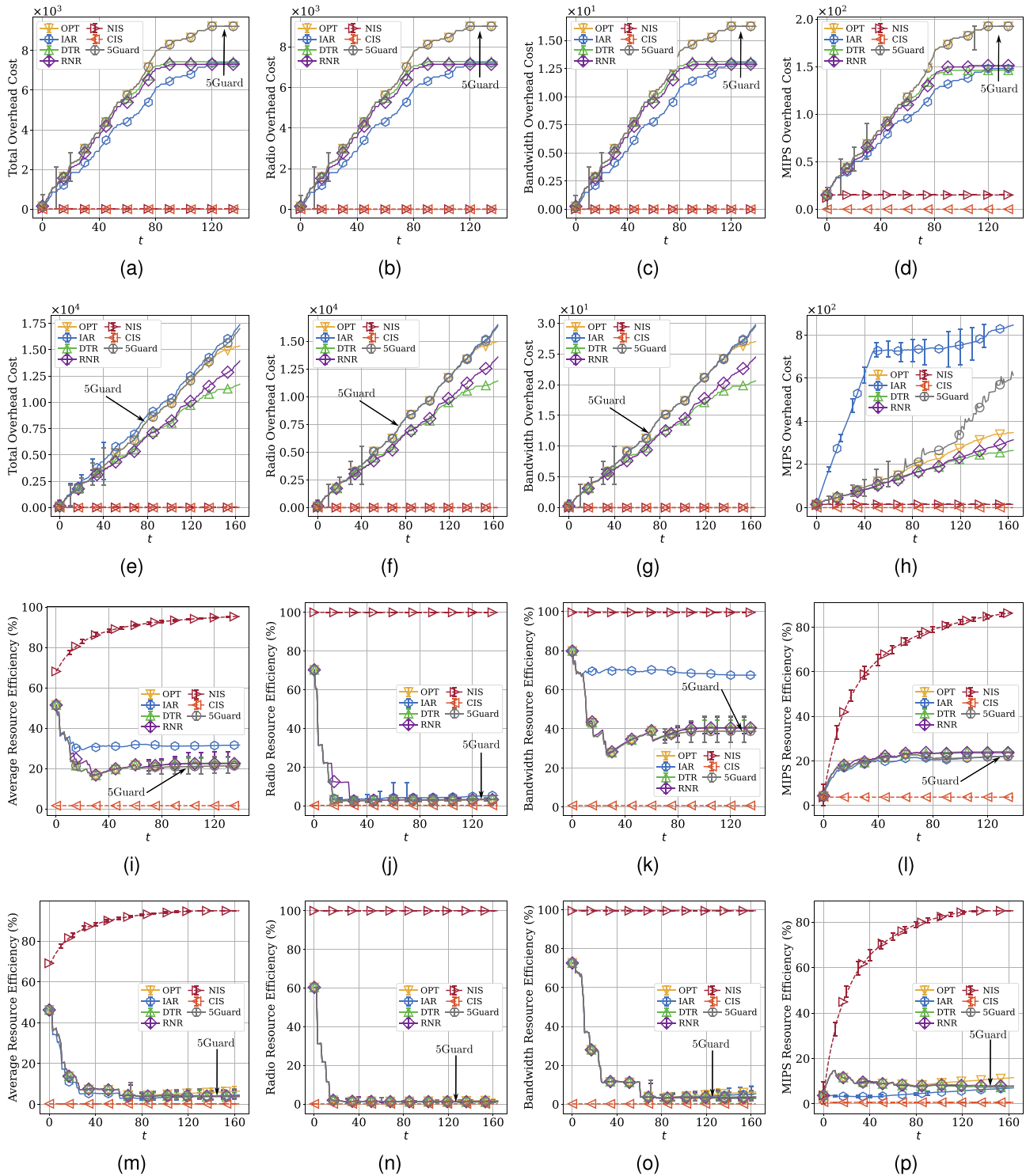


Fig. 9. Comparison of total overhead costs and domain-specific overhead costs along with average resource efficiency and domain-specific resource efficiency considering the isolation overheads and resource usage patterns at each time step for the synthetic (i.e., a–d and i–l) and BRAIN's 5G (i.e., e–h and m–p) topologies.

consumed MIPS is allocated to overheads such as the hypervisor, OpenStack, and host and guest OSs, rather than the actual hosting of NFs. However, under NIS, MIPS efficiency recovers more rapidly due to the ability to share NFs and instantiate fewer containers or VM instances, ultimately leading to higher resource efficiency.

Importantly, isolation-aware approaches enable us to meet SLA requirements while achieving significantly greater efficiency than CIS, particularly for bandwidth and MIPS resources. The difference between isolation-aware methods and CIS for radio resources becomes more evident in scenarios where NR-BS bandwidth is higher.

In contrast to the synthetic topology, where a maximum of only two L2 SRs could be admitted, the BRAIN's 5G topology allows for a significantly higher number of L2 SR admissions. This leads to resource efficiency percentages converging more closely to CIS, especially for radio resources.

To provide a quantitative comparison with existing works, we consider the configuration presented in [16], which assumes two types of slices, MC and non-MC. The setup proposed in [16] assumes both MC and non-MC slices' NFs are deployed with L1 isolation using dedicated VMs. This design incurs a guest OS overhead of  $C_{\text{OVH}}^{\text{GOS}} = 25,000$  MIPS per slice. With  $\Xi_{p,1}^{\text{max}} = 60$ , the total overhead for 60 slices becomes  $25,000 \times 60 = 1,500,000$  MIPS per PS. In contrast, 5Guard assigns L0 isolation to non-MC slices and allows container-level deployment. With  $\Xi_{p,0}^{\text{max}} = 110$  [45] and each slice comprising 5 containers (e.g., corresponding to DU, CU, AMF, UPF, and SMF), the 60 slices require  $60 \times 5 = 300$  containers, which can be hosted on just 3 VMs. This results in a total overhead of  $3 \times (25,000 + 25,000) = 150,000$  MIPS, considering  $C_{\text{OVH}}^{\text{K8S}} = 25,000$ , reducing the overhead to only 10% of that in [16]. This also limits the maximum number of non-MC slices per PS to 60 in [16], while 5Guard supports up to  $60 \times 110/5 = 1,320$  slices under the same configuration.

Additionally, while [16] assumes L2 isolation for transport resources, Fig. 9(k) demonstrates that such isolation can reduce efficiency by up to 97% compared to L0, further highlighting poor scalability for utilizing complete isolation approaches for all slices, similar to what we studied for CIS. Furthermore, the system in [16] relies solely on PRB allocation for controlling performance and considers security in its isolation strategy for future work. Their best-effort non-MC slice experiences a 50% degradation in bandwidth, from 50 MHz to 25 MHz, when an MC slice is added. In contrast, our system enables performance guarantees even for slices using L0 isolation, while maintaining significantly lower overhead, utilizing priority-controlled QoS violation costs.

### C. Key Remarks and Limitations

As this work aims to study the impact of incorporating isolation levels into 5G NS, 5Guard operates under the assumption that the InP admits all SRs whenever the constraints permit. Consequently, 5Guard does not include a proactive admission control mechanism. Admitting all SRs can influence the overall admission percentage and the long-term profit of the 5G InP. However, a critical remark about 5Guard is that, since it maximizes online profit at each time step, it can be well integrated with reward-based admission control schemes, where the 5Guard output serves as a reward signal. To handle tight resource constraints, especially when L2 isolation is requested, 5Guard exploits priority-controlled migration and QoS violation, as well as revisitation mechanisms. Specifically, it allows for controlled QoS violations of LP SRs when necessary to prioritize HP SRs within the link budget. This adaptation makes the provisioning of L2 isolation more feasible in constrained environments. For tight deadlines, as shown in Fig. 8, 5Guard switches to faster algorithms in the ensemble.

To assess the practicality of 5Guard, we address its integration and scalability in real-world deployments. 5Guard operates at the management and orchestration (MANO) layer and can interface with existing MANO platforms, such as ONAP's Service Orchestrator [46], via RESTful APIs. In terms of scalability, the number of algorithms in the ensemble, i.e.,  $|\mathcal{A}|$ , is fixed (e.g.,  $|\mathcal{A}| = 4$  in this work) and can be distributed across dedicated CPU cores, which are readily available in server-grade processors (e.g., AMD EPYC 9684X [42] with 96 physical cores). As described in Algorithm 1, this results in minimal coordination overhead, allowing the framework to remain scalable even under strict deadline constraints. Moreover, Python tools such as "psutil" [47] can be used to pin processes to specific cores, and modern solvers, e.g., Gurobi [38], natively support multi-threaded execution to accelerate computations further.

Another remark is that optimization solvers usually aim to utilize the available CPU cores fully to minimize execution time. As a result, we studied execution times rather than CPU utilization, which may not be a meaningful measure of overhead or scalability.

## VII. CONCLUSION AND FUTURE WORK

This paper explored the theoretical and practical implications of incorporating isolation into 5G network slicing. We formulated the isolation-aware 5G network slicing problem and studied its complexity. We then proposed a novel adaptive framework, 5Guard, which solves the problem by leveraging an ensemble of algorithms ranging from fast to exact methods, selected based on resource allocation deadlines. Next, we evaluated 5Guard's efficacy in challenging scenarios and large-scale networks through simulation. Finally, we analyzed the impact of isolation on resource overhead and efficiency for different algorithms.

For future work, we will investigate the impact of sequential learning-based admission control on the long-term performance of 5Guard. Another direction is to design a block that intelligently selects the appropriate set of algorithms for the ensemble in 5Guard. Finally, we aim to develop solutions for translating security requirements into isolation levels.

## REFERENCES

- [1] P. Promponas and L. Tassioulas, "Network slicing: Market mechanism and competitive equilibria," in *Proc. IEEE Conf. Comput. Commun.*, New York City, NY, 2023, pp. 1–10.
- [2] A. Majidi, N. Jahanbakhsh, and S. Clarke, "Time-varying real-time online multi-resource allocation for scaling the slices and VNF isolation," in *Proc. IEEE Conf. Comput. Commun. Workshops*, Hoboken, NJ, 2023, pp. 1–6.
- [3] J. Liu, B. Zhao, M. Shao, Q. Yang, and G. Simon, "Provisioning optimization for determining and embedding 5G end-to-end information centric network slice," *IEEE Trans. Netw. Service Manag.*, vol. 18, no. 1, pp. 273–285, Mar. 2021.
- [4] P. Promponas and L. Tassioulas, "5G-WAVE: A core network framework with decentralized authorization for network slices," in *Proc. IEEE Conf. Comput. Commun.*, Vancouver, BC, Canada, 2024, pp. 2308–2317.
- [5] *Security Considerations for 5G Network Operation*. Düsseldorf, Germany: NGMN Alliance, 2021.
- [6] Y. He et al., "Cross container attacks: The bewildered eBPF on clouds," in *Proc. USENIX Secur. Symp.*, Anaheim, CA, 2023, pp. 5971–5988.

- [7] *6G Trustworthiness Considerations V1.0*. Düsseldorf, Germany: NGMN Alliance, 2023.
- [8] *3GPP TS 28.530 Release 18: Management and Orchestration; Concepts, Use Cases and Requirements*. Sophia Antipolis, France: 3rd Generation Partnership Project, 2023.
- [9] *5G Mobile Network Sharing Security V1.02*. Düsseldorf, Germany: NGMN Alliance, 2022.
- [10] N. Yarkina, A. Gaydamaka, D. Moltchanov, and Y. Koucheryavy, "Performance assessment of an ITU-T compliant machine learning enhancements for 5G RAN network slicing," *IEEE Trans. Mobile Comput.*, vol. 23, no. 1, pp. 719–736, Jan. 2024.
- [11] H. Yu, F. Musumeci, J. Zhang, M. Tornatore, and Y. Ji, "Isolation-aware 5G RAN slice mapping over WDM metro-aggregation networks," *J. Lightw. Technol.*, vol. 38, no. 6, pp. 1125–1137, Mar. 2020.
- [12] C.-Y. Chang et al., "Slice isolation for 5G transport networks," in *Proc. IEEE 7th Int. Conf. Netw. Softwarization*, Tokyo, Japan, 2021, pp. 366–368.
- [13] A. J. Gonzalez et al., "The isolation concept in the 5G network slicing," in *Proc. Eur. Conf. Netw. Commun.*, Dubrovnik, Croatia, 2020, pp. 12–16.
- [14] R. Khan, P. Kumar, D. N. K. Jayakody, and M. Liyanage, "A survey on security and privacy of 5G technologies: Potential solutions, recent advancements, and future directions," *IEEE Commun. Surv. Tut.*, vol. 22, no. 1, pp. 196–248, First Quarter 2020.
- [15] H. Cheng et al., "ORANSlice: An open source 5G network slicing platform for O-RAN," in *Proc. 30th Annu. Int. Conf. Mobile Comput. Netw.*, Washington D.C., DC, 2024, pp. 2297–2302.
- [16] X. Limani et al., "Optimizing 5G network slicing: An end-to-end approach with isolation principles," in *Proc. IEEE Conf. Netw. Funct. Virtualization Softw. Defined Netw.*, Natal, Brazil, 2024, pp. 1–6.
- [17] A. Lekidis, "Towards 5G advanced network slice assurance through isolation mechanisms," in *Proc. 19th Int. Conf. Availability, Rel. Secur.*, Vienna, Austria, 2024, pp. 1–7.
- [18] C. De Alwis, P. Porambage, K. Dev, T. R. Gadekallu, and M. Liyanage, "A survey on network slicing security: Attacks, challenges, solutions and research directions," *IEEE Commun. Surv. Tut.*, vol. 26, no. 1, pp. 534–570, First Quarter 2024.
- [19] V. Del Piccolo, A. Amamou, K. Haddadou, and G. Pujolle, "A survey of network isolation solutions for multi-tenant data centers," *IEEE Commun. Surv. Tut.*, vol. 18, no. 4, pp. 2787–2821, Fourth Quarter 2016.
- [20] Z. Kotulski, T. W. Nowak, M. Sepczuk, and M. A. Tunia, "5G networks: Types of isolation and their parameters in RAN and CN slices," *Elsevier Comput. Netw.*, vol. 171, Apr. 2020, Art. no. 107135.
- [21] D. Sattar and A. Matrawy, "Towards secure slicing: Using slice isolation to mitigate DDoS attacks on 5G core network slices," in *Proc. IEEE Conf. Commun. Netw. Secur.*, Washington, DC, 2019, pp. 82–90.
- [22] A. Javadpour, F. Ja'fari, T. Taleb, and C. Benzaid, "Reinforcement learning-based slice isolation against DDoS attacks in beyond 5G networks," *IEEE Trans. Netw. Service Manag.*, vol. 20, no. 3, pp. 3930–3946, Sep. 2023.
- [23] P. Zhang, C. Wang, N. Kumar, W. Zhang, and L. Liu, "Dynamic virtual network embedding algorithm based on graph convolution neural network and reinforcement learning," *IEEE Internet Things J.*, vol. 9, no. 12, pp. 9389–9398, Jun. 2022.
- [24] C. L. Moreira, C. A. Kamienski, and R. A. Bianchi, "5G and edge: A reinforcement learning approach for virtual network embedding with cost optimization and improved acceptance rate," *Elsevier Comput. Netw.*, vol. 247, Jun. 2024, Art. no. 110434.
- [25] H. Li, L. Wang, Z. Zhu, Y. Chen, Z. Lu, and X. Wen, "Multicast service function chain orchestration in SDN/NFV-enabled networks: Embedding, readjustment, and expanding," *IEEE Trans. Netw. Service Manag.*, vol. 20, no. 4, pp. 4634–4651, Dec. 2023.
- [26] S. S. Johari et al., "DRL-assisted reoptimization of network slice embedding on EON-enabled transport networks," *IEEE Trans. Netw. Service Manag.*, vol. 20, no. 1, pp. 800–814, Mar. 2023.
- [27] D. Basu, S. Kal, U. Ghosh, and R. Datta, "DRIVE: Dynamic resource introspection and VNF embedding for 5G using machine learning," *IEEE Internet Things J.*, vol. 10, no. 21, pp. 18971–18979, Nov. 2023.
- [28] D. M. Manias, I. Shaer, J. Naoum-Sawaya, and A. Shami, "Robust and reliable SFC placement in resource-constrained multi-tenant MEC-enabled networks," *IEEE Trans. Netw. Service Manag.*, vol. 21, no. 1, pp. 187–199, Feb. 2024.
- [29] R. Zhu, G. Li, Y. Zhang, Z. Fang, and J. Wang, "Load-balanced virtual network embedding based on deep reinforcement learning for 6G regional satellite networks," *IEEE Trans. Veh. Technol.*, vol. 72, no. 11, pp. 14631–14644, Nov. 2023.
- [30] *E2E Network Slicing Architecture V2.0 NG.127*. London, U.K.: GSM Association, 2022.
- [31] *3GPP TS 23.501 release 19: System architecture for the 5G System (5GS)*. Sophia Antipolis, France: 3rd Generation Partnership Project, 2025.
- [32] 3rd Generation Partnership Project (3GPP), "5G critical communications & application enablement for verticals," 2024. [Online]. Available: <https://www.3gpp.org/technologies/sa6-cc-apps>
- [33] *3GPP TS 22.280 Release 19: Mission Critical Services Common Requirements (MCCoRe)*. Sophia Antipolis, France: 3rd Generation Partnership Project, 2023.
- [34] 3rd Generation Partnership Project (3GPP), "5G system overview," 2022. [Online]. Available: <https://www.3gpp.org/technologies/5g-system-overview>
- [35] *3GPP TR 26.985 Release 18: Vehicle-to-Everything*. Sophia Antipolis, France: 3rd Generation Partnership Project, 2024.
- [36] *3GPP TS 38.104 Release 18: NR; Base Station (BS) radio transmission and reception*. Sophia Antipolis, France: 3rd Generation Partnership Project, 2024.
- [37] *ITU G.698.2: Amplified Multichannel Dense Wavelength Division Multiplexing Applications With Single Channel Optical Interfaces*. Sophia Antipolis, France: International Telecommunication Union, 2018.
- [38] Gurobi Optimization, "The state-of-the-art mathematical programming solver," Accessed: Jul. 17, 2024. [Online]. Available: <https://assets.gurobi.com/pdfs/Gurobi-Corporate-Brochure.pdf>
- [39] Y. Shiloach and U. Vishkin, "An  $O(n^2 \log n)$  parallel max-flow algorithm," *J. Algorithms*, vol. 3, no. 2, pp. 128–146, Feb. 1982.
- [40] A. Nemirovski, *Interior-point polynomial algorithms in convex programming*, Philadelphia, PA, USA: SIAM, 1994.
- [41] E. Rosen, A. Viswanathan, and R. Callon, "Multiprotocol label switching architecture," Internet Eng. Task Force (IETF), Jan. 2001, doi: [10.17487/RFC3031](https://doi.org/10.17487/RFC3031).
- [42] PassMark Softw, "AMD EPYC 9684X - CPU benchmark," 2025, Accessed: Mar. 24, 2025. [Online]. Available: <https://www.cpubenchmark.net/cpu.php?cpu=AMDEPYC9684X&id=5971>
- [43] The Zuse Institute Berlin, "BRAIN - Berlin research area information network," Accessed: Jul. 17, 2024. [Online]. Available: <https://www.brain.de/>
- [44] Y. Zhang, W. Liang, Z. Xu, X. Jia, and Y. Yang, "Profit maximization of delay-sensitive, differential accuracy inference services in mobile edge computing," *IEEE Trans. Mobile Comput.*, vol. 24, no. 7, pp. 6209–6224, Jul. 2025.
- [45] "Considerations for large clusters," 2024. Accessed: Jul. 29, 2025. [Online]. Available: <https://kubernetes.io/docs/setup/best-practices/cluster-large/>
- [46] "Service orchestrator Montreal release," 2023. Accessed: Jul. 31, 2025. [Online]. Available: <https://lf-onap.atlassian.net/wiki/spaces/DW/pages/16521708/ServiceOrchestratorMontrealRelease/>
- [47] Giampaolo Rodola, "Psutil: Cross-platform lib for process and system monitoring in Python," Accessed: Jul. 18, 2025. [Online]. <https://github.com/giampaolo/psutil>



**Mehdi Bolourian** (Graduate Student Member, IEEE) received the BSc degree in electrical and computer engineering from Isfahan University of Technology, Iran, in 2020 and the MSc degree in electrical engineering from the Sharif University of Technology, Iran, in 2022. He is a PhD researcher in computer science with the University of Waterloo. He has industry experience in the Internet of Things and co-founded a startup in smart home appliances. He has served as a TPC member for IEEE Globecom 2025. His research interests include optimization and machine learning for orchestrating 5G/6G, mobile-edge/cloud computing, and quantum key distribution networks.



**Noura Limam** (Member, IEEE) received the MSc and PhD degrees in computer science from the University Pierre and Marie Curie (currently Sorbonne University), France, in 2002 and 2007, respectively. She is a research assistant professor of computer science with the University of Waterloo, Canada. She is an active researcher and a contributor in the area of network and service management. Her current interests revolve around network automation and cognitive network management. She is the chair of the *IEEE ComSoc Network Operations and Management*

*Technical Committee*, an associate editor of *IEEE Transactions on Network and Service Management*, an associate editor of *IEEE Communications Magazine*, and a guest editor of *IEEE Communications Magazine Network Softwarization and Management Series*.



**Raouf Boutaba** (Fellow, IEEE) received the MSc and PhD degrees in computer science from Sorbonne University, in 1990 and 1994, respectively. He is currently a University Chair Professor and the Director of the David R. Cheriton School of Computer science with the University of Waterloo (Canada). He also holds an INRIA International Chair in France. He is the founding editor-in-chief of *IEEE Transactions on Network and Service Management* (2007- 2010). He is a fellow of the IEEE, the Engineering Institute of Canada, the Canadian Academy of Engineering, and the Royal Society of Canada. His research interests include resource and service

management in networks and distributed systems.



**Mohammad Ali Salahuddin** (Member, IEEE) received the PhD degree in computer science from Western Michigan University, in 2014. He is currently a research assistant professor of computer science with the University of Waterloo. His research interests include the Internet of Things, content delivery networks, network softwarization, network security, and cognitive network management. His coauthored research publications have received numerous awards, including the ACM SIGMETRICS Best Student Paper, the IEEE/IFIP NOMS Best Papers, and the IEEE

CNOM Best Paper. He serves on the TPC for international conferences and is a reviewer for various peer-reviewed journals, magazines, and conferences.