Praetorian: Probabilistic Planning for Radio Access Network Slice Assurance

Ajay Kattepur, Swarup Mohalik
Ericsson Research Artificial Intelligence
Bangalore, India.
ajay.kattepur@ericsson.com

Ian Burdick
Ericsson Cognitive Network Solutions
Plano, Texas.

***Abstract*—Radio access network (RAN) slicing has been proposed to efficiently serve multiple services such as mobile broadband, augmented reality, gaming and low latency applications. The dynamic allocation of resources typically make use of physical resource blocks (PRBs) share configurations. However, there have been limited studies on the side-effects and uncertainty of dynamic changes in configurations. In this paper, we present Praetorian, a probabilistic planning approach to analyze RAN slice assurance. Detailed queuing network models are used to extract the probabilistic effects of domain actions. These are fed into probabilistic planners to select optimal actions such as priority changes, PRB share changes or dropping. Planning techniques are proposed that improve the resilience of slice assurance techniques under uncertainties. This approach is applied on a real use case from a mobile network operator requiring slice assurance.**

***Index Terms*—Probabilistic planning, Radio Access Network, Slice assurance, Queuing models.**

I. Introduction

The emergence of 5G new radio [1] has increased focus on multiple use cases including enhanced mobile broadband, low latency applications and eXtended Reality (XR). In order to meet the requirements of differentiated Quality of Service (QoS) over a common physical network, techniques such as *network slicing* [2] have been proposed. Network slicing allows the physical radio access network (RAN), transport network and core network to be virtually partitioned to meet differentiated requirements. The slicing has to be done effectively to meet both service guarantee as well as resource utilizations constraints.

A slicing technique used towards the RAN network is to efficiently allocate Physical Resource Blocks (PRBs). While a static allocation can be instantiated at design time, to meet changing intent [3] and traffic requirements, the slice assurance process must dynamically make changes to the PRB partitions. Specially with priority services that require stringent Service Level Agreements (SLAs), the slices must be continuously monitored and assured to meet intents. Automation techniques using Artificial Intelligence (AI) for slice assurance have been proposed [4] [5].

Previous papers have looked at a deterministic planning approach for 5G slice allocation [6]. However, the uncertainty in performance including changes in PRB allocation, latency and throughput as a result of changes have not been analyzed in detail. This paper presents Praetorian, a probabilistic planning technique [7] to analyze the configurations in RAN slicing. Via the use of queuing network models [8], the probabilistic effects of setting queue priority, dropping and partition changes are analyzed in detail. Slice assurance is performed via AI planning techniques [10] to determine optimal actions. The queuing modeling inputs are critical to ensure that uncertainty or changes are mitigated within the slice assurance process using probabilistic planning techniques. Such detailed modeling is crucial for dynamically varying environments such as RAN. This technique is demonstrated over a real mobile network operator slice specification. The main contributions of this paper include:

- Detailed queuing network model to analyze the effect of RAN partition, priority and packet drop setting.
- Probabilistic planning approach that takes in uncertainty RAN slice configuration into consideration to make more realistic slice assurance towards intents.

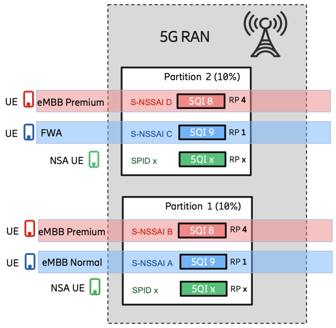
The rest of this paper is organized as follows: Section II presents the RAN slice assurance use case and an overview of Praetorian. Section III presents the queuing network model and analysis for slice resource allocation. An overview of the AI planning formulation is presented in Section IV. The probabilistic planning formulation and evaluation on slice assurance is presented in Section V. Related work is presented in Section VI, followed by conclusions in Section VII.

II. RAN Slice Assurance Use Case

In this section, we describe the RAN slicing use case and associated intent requirements.

A. RAN Slicing

To ensure end-to-end 5G network slicing, one essential network subnet is the Radio Access Network (RAN) [2]. To slice the RAN, an often used technique is allocation of physical resource blocks (PRBs) [5], which is defined as 12 consecutive subcarriers in the frequency domain of the allocated spectrum [1]. The total available PRBs are partitioned and allocated to various slices to meet the 5G QoS Identifier (5QI) [1] requirements. A *PRB partition* is the minimum guaranteed share of radio resources and may be dynamically allocated/deallocated to slices.



	5QI8	5QI9
Priority	High	Low
Throughput (Mbps)	500	300
Packet Latency (ms)	300 ms	300 ms

Fig. 1. RAN Slice Partitioning Scenarios.

Figure 1 presents a RAN slicing scenario motivated by Ericsson's interaction with a mobile network operator from Asia. The network supports four slices with *Premium eMBB Traffic* (5QI8), *Normal eMBB Traffic* (5QI9), Fixed Wireless Access Traffic (5QIx) over two partitions. Partition 1 has eMBB premium slice and eMBB normal slice; Partition 2 has eMBB slice and FWA slice. Both partitions are initially allocated 10% share each of the total available PRBs with the residual pool of 80% PRBs allocated to eMBB traffic outside of the slices.

Initially, the partitions are typically under-utilized; as the user traffic increases, an over-utilized partition will consume any residual resources that are available (Figure 1). With more slices and subscribed users, there is always the risk of Premium SLA violation due to resource contention: in the un-partitioned residual, premium users may not receive the same priority as within exclusive partitions. The slice assurance system should ensure that the premium users' SLAs do not deteriorate, while maintaining fair allocation to normal category users. In order to monitor and assure the RAN slices, the following metrics are typically collected via performance counters:

1) Service Throughput: Average Downlink (DL) Throughput, Average Uplink (UL) Throughput.
2) Resource Utilization: Partition Utilization, Average number of user sessions.
3) Radio Link Quality: Channel Quality Indicator (CQI), Modulation and Coding Scheme (MCS).

The 5G throughput and PRB allocations are done according to [9]. Depending on the spectrum, allocated bandwidth and selected subcarriers, the number of PRB may be calculated. In addition to the number of PRB, the throughput is affected by the modulation scheme. The supported data rate is presented in [9, Sec. 4] and reproduced here (in Mbps) as

$$10^{-6} \cdot \sum_{j=1}^{J} \left(v^{(j)} \cdot f^{(j)} \cdot Q_{\mathrm{m}}^{(j)} \cdot R_{\mathrm{m}} \cdot \frac{12 \cdot N_{\mathrm{PRB}}^{(j)}}{T_\mu} \cdot \left(1 - OH^{(j)}\right) \right) S \tag{1}$$

– J is the number of aggregated carriers,
– $v^{(j)}$ is the number of MIMO layers per carrier,
– $f^{(j)}$ is the scaling factor per carrier in the range $[0.4, 1]$,
– $Q_{\mathrm{m}}^{(j)}$ is the modulation order per carrier,
– R_{m} is the modulation code rate divided by 2048,

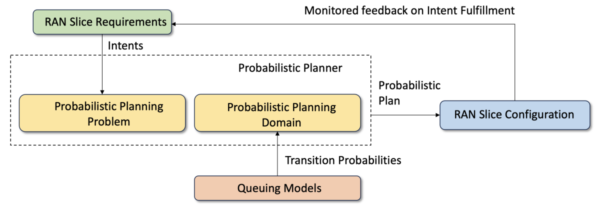


Fig. 2. PRAETORIAN RAN slice assurance framework.

– T_μ is the average OFDM symbol duration in a subframe for numerology μ, calculated as $10^{-3}/(14 \cdot 2^\mu)$,
– $N_{\mathrm{PRB}}^{(j)}$ is the number of physical resource blocks per carrier j, for the given bandwidth and numerology,
– OH is the overhead in range $[0.08, 0.18]$,
– and S is the symbols allocation which determines how much of a slot is dedicated to uplink or downlink.

In typical scenarios, parameters that can be configured are N_{PRB} and the modulation scheme (Q_{m} and R_{m}). Each PRB partition can be configured with a share of resources for which the users belonging to the partition has priority.

B. *RAN Slicing Intents*

In conjunction with the RAN slicing use case, there are intents [3] that are to be managed in an autonomous fashion. These intents include protection of premium service SLAs and dynamic resource management with optimal usage of RAN resources. We specify two tiers of intents:

1) *Service Requirements*: There could be multiple service requirements from the RAN slice services:
   a. Maintain 5QI for premium customers at priority, even with increasing traffic (SLA).
   b. Ensure minimum guaranteed throughput for normal tier customers (fairness).
2) *Radio Resource Management Requirements*: Efficient PRB management to ensure optimal spectrum use, that is minimize unused PRBs in allocated partitions.

Intent violations may be caused due to new requirements or a spike in traffic. The assurance system can automatically reconfigure to mitigate the violations. However, the changes to the RAN system (priorities, partition sizes, drop rates) are not deterministic, causing side-effects to other users of the system. To analyze the effect of this model uncertainty, we introduce the PRAETORIAN framework.

C. *PRAETORIAN Framework*

As presented in Figure 2, the PRAETORIAN framework consists of five components:

1) We start with the state and action models of the domain. The states reflect the throughput and latency values of each slice resulting in intent satisfaction/violation. Actions that can taken are changes in PRB allocation, service priority and drop rates.
2) The queueing model is a simulator for multiple slices that can be used to compute the probabilities for the Probabilistic Planning Domain Definition Language

TABLE I
Queueing network metrics.

V_i	Average number of times packet visits resource i
S_i	Mean service time per packet at resource i
U_i	Utilization of resource i
Q_i	Queue length at resource i
X_i	Throughput of resource i
X	Throughput of the system
D_i	Service demand of resource i
N	Average number of packets in the system
R	Average response time of the system
Z	Mean think time of a terminal user

(PPDDL) domain actions. The evaluated outputs are used to set the transition probabilities.

3) RAN slice specifications that are processed as intents [3] within the planning formulation. The intent specifications are used to generate the problem files (goals) for the planning formulation.
4) Probabilistic planners are used to generate optimal configurations to assure RAN slices. The current state is the initial state - it is the state where an intent is violated and triggers the planning algorithm.
5) The plan configuration executed and monitored for RAN intent satisfaction.

Further details of the queuing models, AI planning formulation and probabilistic plans are presented in the following sections.

III. queueing

A. Queueing Model

Queueing network models have been used to perform performance modeling and analysis of computer systems and networks. Fundamental laws applicable to queuing networks have been proposed using the metrics in Table 1. We briefly review them; an interested reader is referred to [8] for further details.

– Utilization Law: Utilization U is the fraction of time the resource is busy and is dependent on throughput X and service times S. Resources with high utilization cause bottlenecks.

$$U_i = X_i \cdot S_i \tag{2}$$

– Service Demand Law: Total average service time required by a packet at resource i, denoted D_i is dependent on the visits V_i and service times S_i.

$$D_i = V_i \cdot S_i = \frac{U_i}{X} \tag{3}$$

– Little's Law: If there are N users in the system, each with *think times* Z (time waiting between interactions with the system) and the throughput rate X producing a wait time R, the following relationship applies:

$$N = X \cdot (R + Z) \tag{4}$$

Mean value analysis (MVA) [8] has been applied with considerable success in the case of closed queuing networks in order to predict performance at higher work loads. In the single class case, the exact MVA Algorithm starts with an empty network; it then increases the number of packets by 1 at each iteration until there are the required number (N) of packets in the system. For each queuing station $k = 1, ..., K$, the waiting time R_k is computed using the static input service demands S_k and the number of jobs in the queue Q_k. The system throughput is then computed using the sum of waiting times at each node and Little's law (eq. 4). This process is slightly more involved for the case with multiple classes say $1, ...C$. Algorithm 1 presents the MVA technique for multi-class queues. The population of each of the classes are increased proportionally.

Algorithm 1: Mean Value Analysis of Multi-Class.

Input: queueing stations, traffic distribution.
Output: throughput, latency, utilization.
for $k \leftarrow 1$ *to* K **do**
  Set $Q_k^0 \leftarrow 0$; `// Initializing queue length`
for $n \leftarrow 1$ *to* $\sum_{c=1}^{C} N_c$ **do**
  for population $n = (n_1, \dots n_c$ **do**
    for $c \leftarrow 1$ *to* C **do**
      for $k \leftarrow 1$ *to* K **do**
        $R_{c,k} \leftarrow D_{c,k} \left[1 + Q_k^{(n-1)c}\right]$; `// Latency estimate with increasing queue length`
      $X_c \leftarrow \frac{n_c}{z_c + \sum_{k=1}^{K} R_{c,k}}$; `// Throughput estimate`
    for $k \leftarrow 1$ *to* K **do**
      $Q_k^n \leftarrow \sum_{c=1}^{C} X_c R_{c,k}$; `// Queue length update`
Return throughput and latency per class;

B. Tools for Queue Analysis

In order to study the queueing network model of the RAN resources, we make use of the Java Modeling Tools[1] simulator. We present a few salient features that we make use of in the RAN slice assurance model:

– Queueing Station: The arriving packets join the queue and wait to receive service from the first idle server.
– Multiple class models consist of C classes, with varying traffic patterns and service demand at each station. Packets are ordered according to their arrival time but packets with higher priority jump ahead of packets with lower priority.
– Routing Station: In the routing section, for each class, the generated packets are routed to the devices connected to the analyzed station according to various routing strategies. The routing probability for each outgoing link must be defined.

Figure 3 presents the closed queuing model of the RAN slice assurance use case. Here, each queue represents 10% of the partition shares with a routing station used to divert traffic in proportion. Multiple classes of services are simulated according to Figure 1, with priorities. Using Mean Value Analysis techniques, the queueing models may perform what-if analysis to estimate performance with various traffic mixes.

[1] https://jmt.sourceforge.net

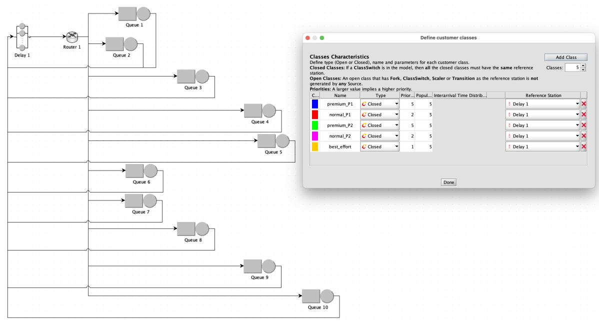


Fig. 3. Queueing model for slice assurance.

IV. AI Planning

AI Planning [10] focuses on composing a set of actions to reach a goal state. AI Planning begins with the definition of domains, plans and goals that are to be achieved.

Definition 1: **Planning Domain** A planning domain is a state transition system $\Sigma = (S, A, \gamma, \mathcal{C})$, where:

- S is a finite set of states of the system.
- A is a set of actions that may be performed by an agent.
- $\gamma : S \times A \rightarrow S$ is the state transition function. If $\gamma(s, a)$ is defined, than action a is applicable to state s, with $\gamma(s, a)$ being the predicted outcome.
- $\mathcal{C} : S \times A \rightarrow [0, \infty)$ is a cost function with the same domain as γ. It can represent a cost function minimizing monetary cost, latency or parameters within the system.

Definition 2: **Plan** A plan is a finite set of actions:

$$\pi = \langle a_1, a_2, \ldots, a_n \rangle$$

where the plan's length $|\pi| = n$ and cost is $\mathcal{C}(\pi) = \sum_{i=1}^{n} a_i$.

Definition 3: **Planning Problem** A planning problem is specified as a triple $P = (\Sigma, s_0, g)$ where Σ is a planning domain, s_0 is the initial state and g is a set of ground literal goals. A solution for P is a plan $\pi = \langle a_1, a_2, \ldots, a_n \rangle$ such that $\gamma(s_0, \pi)$ satisfies g.

Solutions to the classical planning problem may be developed using forward-search or backward-search techniques, with multiple heuristics proposed to reduce the state space search [10]. The Planning Domain Definition Language (PDDL) [11] [10] is an action centered language that provides a standard syntax to describe actions and the pre-conditions/effects of the actions. It consists of two descriptions (i) the *domain* description that decouples the parameters of actions from specific objects, initial conditions and goals (ii) the *problem* description that instantiates a grounded problem with objects, initialization, goals and metrics. The same domain description may be paired with multiple problem instances, with varying grounded objects, initial conditions and goals.

While conventional PDDL has only deterministic action effects, real-world deployments typically have uncertainties in observed outputs. In order to define decision theoretic planning problems, we need to add support for probabilistic effects. The syntax for PPDDL effects is [7] specified:

$$\texttt{probabilistic}\ p_1\ e_1\ p_2\ e_2 \ldots p_k\ e_k$$

meaning that effect e_i occurs with probability p_i. We require that the constraints $\sum_{i=1}^{k} p_i = 1$ are fulfilled: a probabilistic effect declares an exhaustive set of probability-weighted outcomes. For example, the effect `(probabilistic 0.9 (compliant))` means that with probability 0.9 the state variable compliant becomes true in the next state, while with probability 0.1 the state remains unchanged.

V. Probabilistic Planning for Slice Assurance

We initialize the planning problem with the slices presented in Figure 1. The two premium slices have latency and throughput violated with a goal being intent compliance (throughput targets, spectrum efficiency). To study the effects of changing priority, drop and partition share changes on various metrics, the queue model from Figure 3 is simulated in JMT. The multi-class MVA in Algorithm 1 is used to estimate the changes in throughput, latency and queue utilization with increase in traffic.

As presented in Figures 4 and 5, increase in user traffic coupled with configuration changes such as priority or partition share changes can affect services differently. We elaborate further on the following:

– Figures 4 and 5 (a) showcase the increase in queue utilization with increased traffic. We notice that the partition share case causes quicker queuing bottlenecks compared to priority.
– Figures 4 and 5 (b) showcase the throughput for premium and normal slices, that would be affected by the configurations and increased traffic (note the drop in throughput of normal services). Higher throughput is observed for premium services with partition share change compared to priorities.
– Figures 4 and 5 (c) showcase the latency computed via Mean value analysis with corresponding throughput.

The premium eMBB services receive higher throughput (coupled with lower latency) due to higher priority and partition shares. The values are used to populate the transition probabilities within probabilistic planners. Repeated experiments may be conducted on changes in priorities or partition sizes. Note that we assume high MCS index for users throughout the slice.

Table II provides an output of the probabilistic PDDL formulation. Lines 1–11 presents an example of the problem file that is with violated throughput and latency. As further seen in Table II lines 13–25, a PPDDL domain file with actions such as change in priority, drop rates and partition shares. Probabilities may be generated using Bayes rules from the queueing network models outputs of Figures 4 and 5. The problem specification has values for throughput/latency compliance and violation using the `probabilistic` specification. The objective of the planner would be to then maximize the probability of reaching the goal state. To solve the probabilistic formulation, we make

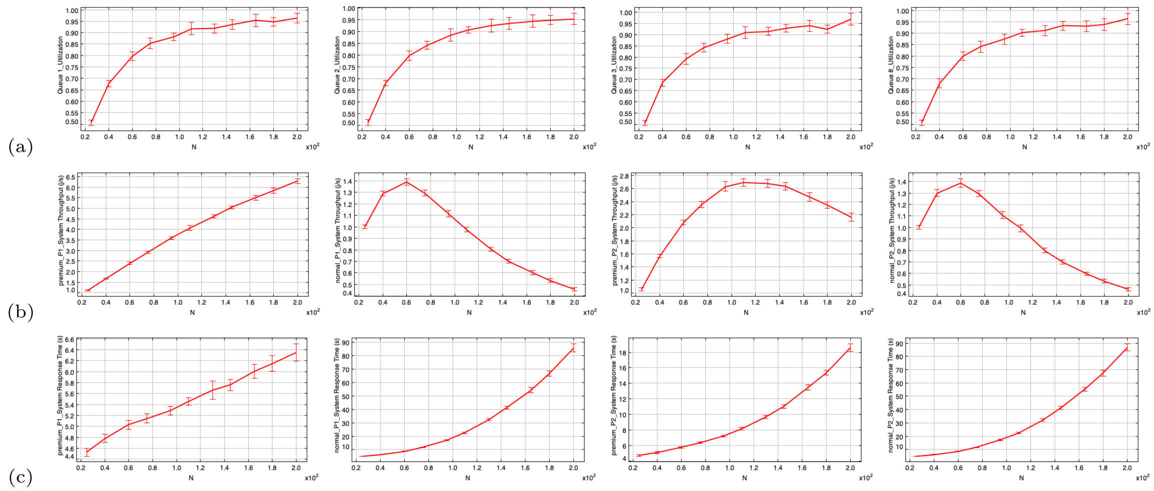


Fig. 4. Priority configuration with resulting changes in (a) Queue Utilization (b) Service Throughput (c) Service Latency.

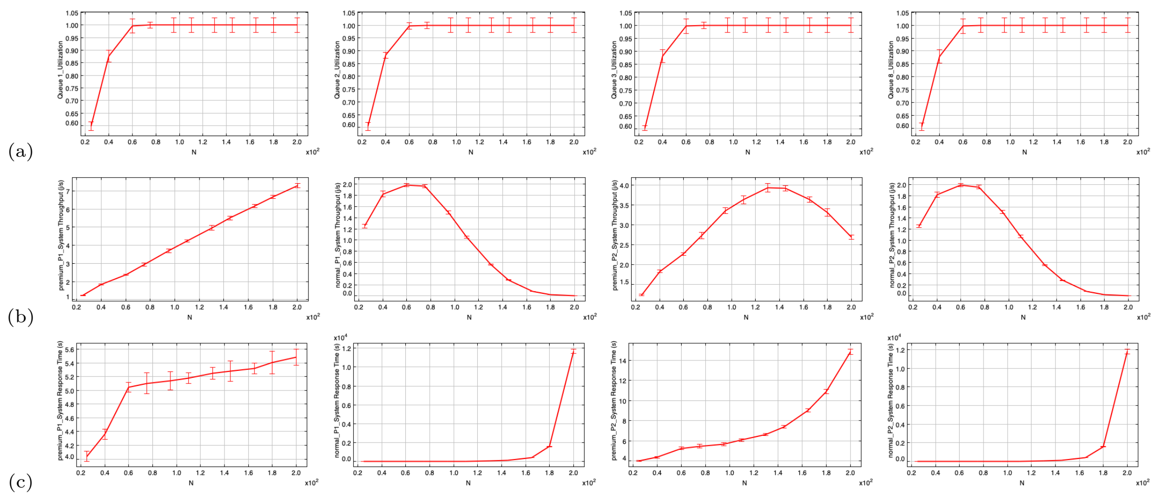


Fig. 5. Partition share configurations with resulting changes in (a) Queue Utilization (b) Service Throughput (c) Service Latency.

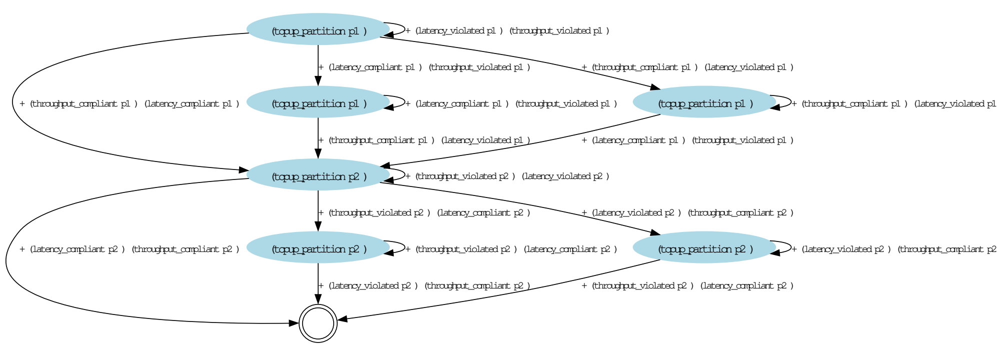


Fig. 6. Probabilistic plan output for RAN slice assurance.

use of the safe planner [12], a planner that can handle PPDDL formulation. An example of the probabilistic plan is presented in Table II lines 27–33 providing the required steps needed to reach the goal state.

As further illustrated in Figure 6, the latency and throughput requirements are evaluated based on the actions taken such as `topup partition`. This process ensures that the non-deterministic effects of changing RAN partition shares or priorities are included within the slice assurance models. Unlike deterministic planning techniques, the probabilistic planning formulation takes into account the uncertainty in actions.

VI. Related Work

AI Driven Slice assurance: As specified in [4], AI techniques for 5G network slicing may be used for: (i)

TABLE II
PROBABILISTIC PDDL SPECIFICATION.

```
; PDDL Problem file snippet
(:init
(throughput_violated eMBB_premium_P1 n5 P1)
(latency_violated eMBB_premium_P1 n5 P1)
(throughput_violated eMBB_premium_P2 n5 P2)
(latency_violated eMBB_premium_P2 n5 P2))

(:goal (and (throughput_compliant eMBB_premium_P1 n5
      P1 )
     (latency_compliant eMBB_premium_P1 n5 P1 )
     (throughput_compliant eMBB_premium_P2 n5 P2 )
     (latency_compliant eMBB_premium_P2 n5 P2 ))))

; Probabilistic PDDL Domain file snippet
(:action change_partition_share
  :parameters (?slice - slice ?band - G5_band ?
       partition - partition)
  :precondition
     (and
     (throughput_violated ?slice  ?band  ?partition)
     (latency_violated ?slice  ?band  ?partition)
  :effect
     (and
     (probabilistic 0.67 (throughput_violated ?slice
         ?band  ?partition)
     0.33 (throughput_compliant ?slice  ?band  ?
         partition))
     (probabilistic 0.54 (latency_violated ?slice ?
         band  ?partition)
     0.46 (latency_compliant ?slice  ?band  ?
         partition))))

; Probabilistic Plan Output snippet
0 : {(topup_partition embb_premium_p1 n5 p1)}
1 : {(topup_partition embb_premium_p1 n5 p1)}
2 : {(topup_partition embb_premium_p1 n5 p1)}
3 : {(topup_partition embb_premium_p2 n5 p2)}
4 : {(topup_partition embb_premium_p2 n5 p2)}
5 : {(topup_partition embb_premium_p2 n5 p2)}

Compilation time: 0.082 s [65 domains]
```

Demand Forecasting: learning user behavior; (ii) Infrastructure: learning how the underlying infrastructure reacts or limits elastic management; (iii) Requirements: service descriptions that can be processed by the network management functionality. In [5], a multi-objective radio resource slice management technique is proposed, which makes use of model-based reinforcement learning techniques to dynamically modify PRB partitions. The AI system is trained to meet multiple objectives such as network slice Service Level Agreement (SLA) compliance, spectrum usage efficiency and fairness among customer classes. In [13], a PRB allocation method for Adaptive RAN is proposed that can improve the communication quality while reducing the information volume.

Queuing models for slicing: In [14], the transport network slicing is modeled internal ingress and egress queues within routers via a queuing model. The effects of changing queue configuration with respect to priority, weights, flow limits, and packet drops are studied in detail. This is used to train a model-based Reinforcement Learning (RL) algorithm to generate optimal policies for flow prioritization, fairness, and congestion control. In [15] constraints on user transmission rate quality of service (QoS) for different kinds of slice SLAs in dynamic heterogeneous RAN scenario with differentially covered RAN slicing.

AI planning techniques: The use of temporal planning techniques has been explored in [6] to jointly plan, schedule and reconfigure robotic tasks in conjunction with appropriate network slicing. In [12], Safe-Planner (SP), an off-line non-deterministic planning algorithm based on Fully Observable Non-Deterministic (FOND) planning problems are proposed. However, these techniques have not been applied to the RAN slicing problem space.

PRAETORIAN: While research directions within slice assurance have been attempting to use AI planning techniques, the variability in changes in partitions, priorities and drop rates have not been sufficiently explored. Detailed queuing models enable analysis of various scenarios that may be fed into AI planners. In addition, the technique is applied to a real-world use case to demonstrate the solution effectiveness.

VII. CONCLUSIONS

Effective slicing of the Radio Access Network is needed to meet the differentiated service requirements of 5G. However, the modeling of the slice configuration has previously not taken the uncertainty in dynamic RAN configurations into account. In this paper, we propose PRAETORIAN, a probabilistic framework for slice assurance. Queueing model outputs are used to model the probabilistic actions. Then, probabilistic planners are used to determine the optimal plan to reconfigure RAN slices. This process improves the resilience of slice assurance, specially with multiple service priorities. The process is demonstrated over a real mobile network use case.

REFERENCES

[1] ETSI, “System architecture for the 5g system”, *TS 23.501*, 2018.

[2] X. Foukas, A. Elmokashfi, G. Patounas, and M. Marina, “Network slicing in 5g: Survey and challenges”, *IEEE Comms. Mag.*, vol. 55, no. 5, pp. 94–100, 2017.

[3] TM Forum, “Intent in Autonomous Networks,” *IG 1253*, 2021.

[4] D. Gutierrez-Estevez et al., ”Artificial Intelligence for Elastic Management and Orchestration of 5G Networks,” *IEEE Wireless Comms.*, vol. 26, no. 5, pp. 134-141, October 2019.

[5] A. Kattepur, S. David and S. Mohalik, ”MUESLI: Multi-objective Radio Resource Slice Management via Reinforcement Learning,” *IEEE NetSoft*, pp. 133-138, 2022.

[6] A. Kattepur and S. Mohalik, ”AI Planning for Tele-operated Robotic Network Slice Reconfiguration,” IEEE 5G World Forum (5GWF), Canada, 2021.

[7] H. Younes and M. Littman, “PPDDL 1.0: An Extension to PDDL for Expressing Planning Domains with Probabilistic Effects”, 2004.

[8] E. Lazowska, “Quantitative System Performance, Computer System Analysis Using Queuing Network Models”, *Prentice Hall*, 1984.

[9] 3GPP, “NR user equipment (ue) radio access capabilities”, *Spec 38.306*, 2022.

[10] M. Ghallab, D. Nau, and P. Traverso, “Automated Planning: Theory and Practice”, *Morgan Kaufmann*, 2004.

[11] M. Ghallab et al., “The planning domain definition language 1.2”, *Yale Center for Computational Vision and Control*, 1998.

[12] V. Mokhtari, A. Sathya, N. Tsiogkas and W. Decré, “Safe-Planner: A Single-Outcome Replanner for Computing Strong Cyclic Policies in Fully Observable Non-Deterministic Domains”, *Intl. Conf. on Advanced Robotics*, pp. 974-981, 2021.

[13] H. Hirayama, Y. Tsukamoto and H. Shinbo, “Feedback Control for QoS-Aware Radio Resource Allocation in Adaptive RAN”, *IEEE Access*, vol. 10, pp. 21563-21573, 2022.

[14] A. Kattepur, S. David and S. K. Mohalik, ”Model-based reinforcement learning for router port queue configurations,” *Intelligent and Converged Networks*, vol. 2, no. 3, pp. 177-197, 2021,

[15] H. Shen, Y. Ye and T. Zhang, ”Joint Slice Level and Base Station Level Admission Control in RAN Slicing,” *International Conference on Communication Technology (ICCT)*, China, pp. 418-422, 2022.