TSINGHUA SCIENCE AND TECHNOLOGY
ISSN 1007-0214 13/31 pp171–185
DOI: 10.26599/TST.2024.9010020
Volume 30, Number 1, February 2025

A P4-Based Approach to Traffic Isolation and Bandwidth Management for 5G Network Slicing

Wenji He, Haipeng Yao*, Huan Chang, and Yunjie Liu

Abstract: With various service types including massive machine-type communication (mMTC) and ultra-reliable low-latency communication (URLLC), fifth generation (5G) networks require advanced resources management strategies. As a method to segment network resources logically, network slicing (NS) addresses the challenges of heterogeneity and scalability prevalent in these networks. Traditional software-defined networking (SDN) technologies, lack the flexibility needed for precise control over network resources and fine-grained packet management. This has led to significant developments in programmable switches, with programming protocol-independent packet processors (P4) emerging as a transformative programming language. P4 endows network devices with flexibility and programmability, overcoming traditional SDN limitations and enabling more dynamic, precise network slicing implementations. In our work, we leverage the capabilities of P4 to forge a groundbreaking closed-loop architecture that synergizes the programmable data plane with an intelligent control plane. We set up a token bucket-based bandwidth management and traffic isolation mechanism in the data plane, and use the generative diffusion model to generate the key configuration of the strategy in the control plane. Through comprehensive experimentation, we validate the effectiveness of our architecture, underscoring its potential as a significant advancement in 5G network traffic management.

Key words: network slicing; P4; traffic isolation; bandwidth management; diffusion model

1 Introduction

With the advancement of fifth generation (5G) and Beyond 5G (B5G) technologies, a new paradigm in mobile communications is unfolding. These technologies are marked by a substantial surge in data traffic and an extensive variety of service offerings, posing new challenges in network resources management[1]. To address these complexities, particularly when diverse applications coexist on a shared network infrastructure, leading to issues like bandwidth congestion and variable connectivity quality, network slicing (NS) emerges as a strategic solution[2]. By segmenting the network into distinct slices each tailored for specific services or applications, NS not only enhances resource efficiency but also improves user experiences by reducing latency and offering customized quality of service (QoS)[3].

Building upon the network management challenges brought forth by the surge in data traffic and service variety in 5G and B5G technologies, the integration of software-defined networking (SDN) with network slicing emerges as a crucial innovation[4]. Characterized by its distinctive separation of the control plane from the data plane, SDN facilitates dynamic resource management, optimizing application

• Wenji He, Haipeng Yao, and Yunjie Liu are with School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China. E-mail: hewenji@bupt.edu.cn; yaohaipeng@bupt.edu.cn; liuyj@pmlabs.com.cn.
• Huan Chang is with School of Information and Electronics, Beijing Institute of Technology, Beijing 100081, China. E-mail: changhuan@bit.edu.cn.
* To whom correspondence should be addressed.
Manuscript received: 2023-11-02; revised: 2023-12-25; accepted: 2024-01-18

performance for a diverse range of requirements. However, when combined with network slicing, traditional SDN encounters unresolved challenges, particularly in terms of adaptability and flexibility due to the constraints imposed by vendor-specific hardware, which becomes especially pronounced when addressing the dynamic and varied requirements of 5G network services[5]. The advent of programmable data planes, exemplified by programming protocol-independent packet processors (P4), introduces a new level of programmability across diverse network devices[6]. As a domain-specific language tailored for orchestrating packet-forwarding mechanisms in various networking hardware, P4's unique strength lies in its ability to support NS with high precision, thereby overcoming the adaptability constraints of traditional data planes.

The integration of P4 programming with network slicing marks a significant shift, enabling precise control over various network segments within a single physical infrastructure, addressing the diverse needs of modern network services[7]. NS effectively partitions a physical network into several virtual segments, each optimized for specific requirements. P4's role as a specialized programming language extends this capability, allowing intricate control over packet processing from header modifications to priority-based forwarding. The effectiveness of P4 in supporting network slicing is evident in several key areas. It allows for dynamic resource distribution, tailored processing routines for each slice, and efficient use of hardware resources[8]. Techniques like dynamic table updates and flexible packet modifications facilitated by P4 lead to reduced latency and enhanced service customization[9]. Some researches focus on the meters in P4 for managing network resources effectively, as they enable rate-limiting and bandwidth control within each slice, isolating traffic to prevent resource contention between slices[10]. By incorporating metering, P4 facilitates advanced techniques like dynamic table updates, flexible packet modifications, and priority queuing, all while ensuring each slice operates within its allocated resources, thereby reducing latency and enhancing service customization. However, the static nature of meter settings in P4 can lead to suboptimal resource allocation, as they cannot dynamically adjust to the fluctuating network conditions and traffic patterns, especially in the context of the ever-changing demands of 5G and B5G networks.

To address the limitations of static meter configurations in P4, we advocate the integration of generative artificial intelligence (GAI) algorithms within the control plane. This approach is designed to dynamically predict and optimize the parameters for bandwidth management in P4 meters, significantly enhancing the adaptability and efficiency of network slicing. In the ever-evolving networking sphere, GAI is poised to revolutionize various aspects of network operations[11]. It offers a breadth of capabilities, from facilitating dynamic responses to real-time network conditions to providing predictive insights for informed decision-making. Furthermore, GAI introduces innovative strategies for resource allocation, ensuring optimal network performance. Amongst the generative models, diffusion models stand out for their unique data generation process[12]. These models are trained by gradually adding Gaussian noise to the training data and then learning to reverse this process, effectively denoising the data. In application, this involves taking random noise samples and processing them through the trained denoising mechanism to generate new data instances. This technique capitalizes on the model's ability to reconstruct original data from noise-altered states, offering a powerful tool for data generation and analysis in network environments.

Within the networking optimization, GAI exerts substantial influence across all network facets. Its impact extends from fundamental aspects like content delivery to the intricate architectural configurations of networks. For instance, GAI enhances network adaptability by enabling dynamic adjustments that respond to real-time conditions. It offers predictive insights that support informed decision-making, and it devises strategic resource allocation methods that are crucial for achieving optimal network performance. Diffusion models are categorized as generative models, designed to generate data akin to the data used for their training. Essentially, these models operate by systematically perturbing the training data with successive increments of Gaussian noise and subsequently learning to restore the original data by reversing this noise application. Following the training phase, employing the diffusion models for data generation becomes a straightforward process–by directing randomly sampled noise through the acquired denoising process, the model generates new data instances. This method leverages the model's learned

capacity to reconstruct the original data from the noise-induced representations[13].

In this article, we introduce an innovative network slicing framework driven by P4 programming, which continuously monitors network traffic and performance metrics to establish dedicated, independent channels within the network. The key focus is on utilizing P4's dynamic capabilities to partition network resources, ensuring isolated traffic within each slice and preventing any interference from adjacent slices. Enhancing this framework, we integrate GAI algorithms, particularly diffusion models, into the control plane. This integration enables the system to predict future traffic patterns and service demands, allowing for proactive and real-time adjustments in slice scaling and resource allocation. Such predictive capabilities are transformative, ensuring uninterrupted service during peak traffic periods and avoiding the need for excessive resource provisioning. The major contributions of this paper are summarized as follows.

(1) We introduce P4's programmability to resource management at the data plane level. This encompasses detailed information management and the implementation of innovative meter designs along with token bucket mechanisms. By exploiting the programmable nature of P4, we achieve precise traffic isolation and cater to diverse service requirements with enhanced accuracy.

(2) We present a novel methodology that integrates generative diffusion model within the control plane. This approach leverages the predictive power of AI to anticipate traffic patterns and service demands, enabling dynamic adjustments in network slicing. This predictive approach allows for more efficient and responsive resource allocation, ensuring the network adapts to changing demands while maintaining service quality.

(3) To evaluate the effectiveness of our proposed closed-loop architecture, we develop a comprehensive architecture using P4 meters. This architecture provides a practical experimental pathway to assess the performance and viability of our approach, showcasing the enhanced adaptability and efficiency brought by the integration of GAI.

The rest of this work is organized as follows. In Section 2, we review the pertinent literature, delineating the existing challenges and current methodologies. In Section 3, we formulate the mathematic model of our proposed system and establish the problem. In Section 4, we propose a P4-based approach to traffic isolation and bandwidth management for the network slicing, In Section 5, we design a generative diffusion algorithm for the bandwidth management strategy. In Section 6, simulation results and performance analysis are presented, followed by a summary of our findings and a discussion on potential future work in Section 7.

2 Related Work

In this section, we will discuss related work from the view of resource management of the network slicing, the usage of the programmable data plane, and the generative AI algorithms for network optimization.

2.1 Resource management of the network slicing

Network slicing necessitates sophisticated resource management strategies for effectively distributing computational, storage, and bandwidth resources across its virtual segments. This ensures that each slice is adequately resourced while maintaining overall network efficiency and reliability. In Ref. [14], Bega et al. proposed a deep learning architecture for predicting the capacity needed to meet future traffic demands within a single network slice, taking into account the operators' desire to strike a balance between resource over-provisioning and service request violations. In Ref. [1], Zhang et al. presented a logical architecture for a 5G system built on network slicing and proposed a scheme for managing mobility between different access networks. In Ref. [15], Jošilo et al. put forward a game theory-based architecture aimed at jointly optimizing the dynamic assignment of computational tasks to slices and resource management. They considered a slicing-enabled edge system where the slice resource orchestrator assigns devices to slices and shares radio resources among them, with the objective of maximizing overall system performance. In Ref. [16], Thantharate et al. proposed the utilization of a transfer learning approach to address the complex network load estimation problem in network slicing. Their goal was to promote a fairer and more equitable distribution of network resources. In Ref. [17], Mai et al. proposed to improve the performance of slicing in terms of quality of service, energy efficiency, and reliability by combining the capabilities of deep reinforcement learning based on a migration learning framework. However, much existing research tends to treat network elements as

black-boxed entities with a relatively coarse granularity of control. This indicates the potential for further advancements in more granular and detailed resource management within network slicing.

2.2 Programmable data plane assisted the network slicing

The role of P4 programmable data planes in network slicing has been a focal point in recent research. In Ref. [7], Chen et al. proposed a design of bandwidth management for QoS with SDN and P4-programmable switch based the function of the meter in P4 switch. In Ref. [9], Wang et al. further designed and implemented a TCP friendly meter in the packet processing pipeline of the P4 switch to realize resource control and quality of service assurance for specific flow services. In Ref. [10], Chen et al. designed the programmable switch's meter to flexibly bandwidth-guarantee and manage network slices by isolating different types of traffic in multiple priority queues while setting appropriate storage bucket sizes. While these studies concentrated on the resource management capabilities of P4-based programmable switches, they mainly emphasized bandwidth resources rather than exploiting the full potential of fine-grained resources provided by these switches. In Ref. [8], Hauser et al. proposed P4-programmable targets are capable of network slicing in all proposed variants. They explored the different aspects of inter-tenant interference due to differences in targeting and slicing methods, and proposed hardware-based slicing methods to eliminate these interferences. In Ref. [18], Pinto et al. proposed a hierarchical SDN optical packet survivability solution based on network slicing is proposed to provide different levels of reliability. Slicing is implemented in a P4 programmable ASIC for traffic prioritization and protection switching. Despite these advancements, the majority of existing approaches primarily focused on managing and optimizing bandwidth resources, utilizing programmable data planes, open switches, and fine-grained storage, compute, and transport resources. However, many of these management schemes predominantly targeted the data plane, often neglecting the powerful control and decision-making capacities available in the control plane.

2.3 Generative AI methods for network optimization

The incorporation of GAI methods in network optimization represents a significant leap in the field of network slicing. In Ref. [19], Xu et al. proposed the deployment of mobile AIGC networks via collaborative cloud-edge-mobile infrastructure is proposed to support wider AIGC services. In Ref. [20], Huang et al. proposed a novel diffusion model-based learning approach to dynamically and adaptively generate the network design to cope with the time-varying environments and various service requirements. In Ref. [21], Huang et al. proposed distributed learning paradigms to enable the AIGC in the wireless network supported by harmonious cloud-edge-mobile infrastructures to enhance a broader range of AIGC services. In Ref. [22], Du et al. proposed a novel collaborative distributed diffusion based AIGC framework, which uses the collaboration among devices in wireless networks and optimizes edge computation resource utilization. These advanced AI algorithms, including diffusion models, bring transformative potential in predictive analytics, leading to more efficient and intelligent network management.

3 System Model and Problem Formulation

This section clarifies the mathematical framework of our P4-based network slicing strategies.

3.1 Network architecture

As depicted in Fig. 1, the network architecture is stratified into three distinct layers: Application plane interfaces with user applications, cataloging specific service demands such as bandwidth, latency, and resilience against packet loss. This layer is pivotal in translating user-centric requirements into precise network configurations, forming the nexus between demand and delivery. Control plane orchestrates the network's operational dynamics, leveraging advanced generative AI algorithms to facilitate informed decision-making processes. It is the linchpin that translates high-level service policies into granular, actionable configurations for the data plane, guiding the network's adaptive behavior to align with service-level agreements and optimization objectives. At the foundation lies programmable data plane, the execution layer where data packets are actively processed, managed, and routed. This is facilitated by the intrinsic programmability of P4 switches, which execute the policies formulated by the control plane, embodying the operational instructions in real-time traffic management.

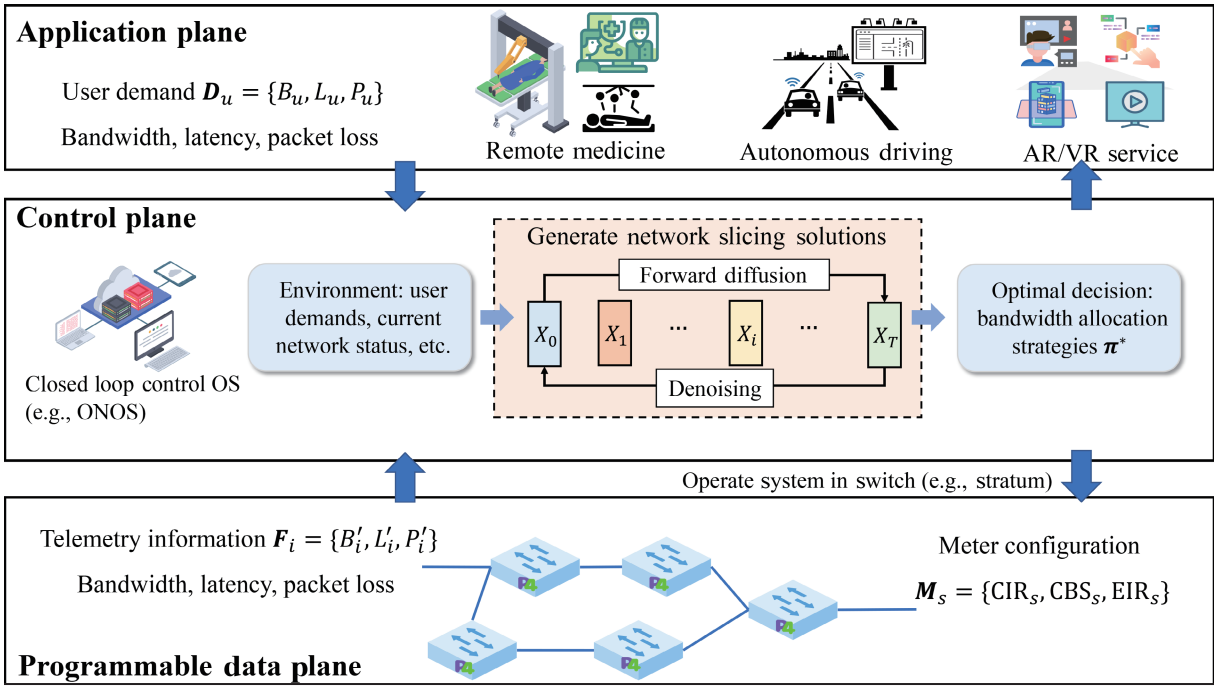

Fig. 1 System model of the P4-based network slicing strategies.

3.2 User model

The user model provides a mathematical characterization of service demands that informs the configuration of network slices to meet the specific QoE objectives for varied 5G applications. We use $D(u) = \{B_u, L_u, P_u\}$ denote the demand of user u as a set that includes bandwidth B_u, latency L_u, and packet loss rate P_u. In the user model, each application's requirements inform the network slice's characteristics. For instance, critical applications such as remote surgery demand a $D(u)$ with high B_u for uninterrupted high-definition video streaming, minimal L_u to ensure real-time responsiveness, and a near-zero P_u for data integrity. In contrast, less critical services like smart home monitoring may present a more lenient set of demands, allowing for higher L_u and a tolerable P_u, reflecting a scalable and flexible approach to resource allocation. These models are essential for tailoring network slices to the demands of 5G services.

3.3 Programmable switch model

Our model for network slicing within a P4 switch environment emphasizes the vital roles of in-band network telemetry (INT) and metering.

3.3.1 INT information

INT is pivotal for providing granular, real-time insights into network performance, tailored to each network slice. As delineated in Fig. 2, the P4 processing pipeline, fundamental to INT, progresses through parsing, match-action decision-making, and packet reassembly, enabling the extraction, processing, and modification of packet data in real-time[6]. This pipeline not only mirrors the network's current state to the control plane but also furnishes the predictive analytics crucial for resource management and network optimization. We denote the INT metadata as $F_i = \{B_i^{'}, L_i^{'}, P_i^{'}\}$, where INT provides real-time data on bandwidth $B_i^{'}$, latency $L_i^{'}$, and packet loss $P_i^{'}$ for each slice i. F_i not only informs the control plane about the current state of the network but also enables predictive analytics for anticipating future network states. These indicators not only reflect the current network state to the control plane but also underpin predictive analytics, essential for forecasting network conditions and preemptively managing resources. Such foresight is integral to proactive resource allocation and traffic shaping. Utilizing INT data, the control plane dynamically orchestrates network slices to meet their respective performance objectives, leveraging sophisticated algorithms for dynamic slicing that ensures each slice upholds its service level without impacting others.

3.3.2 Meter configuration

The meter in P4 is a fundamental construct, functioning to monitor the rate of packet or byte flow over

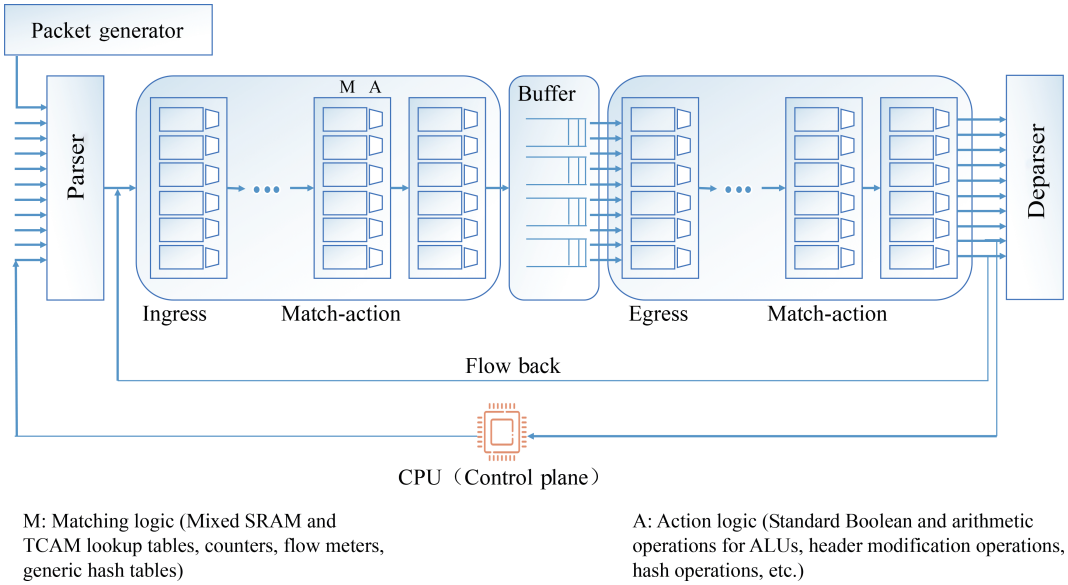

Fig. 2 Architecture of the P4[6].

designated time intervals and to execute specific actions contingent on the measured rates. For slice s, the meter configuration, $\text{Meter}_s = \text{CIR}_s, \text{CBS}_s, \text{PIR}_s\}$, functions as a traffic regulator within P4 switches, aligning traffic flow with control plane policies. CIR_s is the committed information rate, ensuring a consistent bandwidth rate for slice s, which is vital for stable performance and adherence to service requirements. CBS_s represents the committed burst size, allowing slices to maintain performance under typical traffic loads. PIR_s signifies the peak information rate, accommodating maximum bandwidth usage during surge conditions. Effective metering enables the data plane to uphold each slice's service level objectives, including bandwidth limitations and latency targets, while promoting equitable resource sharing among slices. P4 switches' meter configurations are instrumental in realizing these operational controls.

3.3.3 Table entries of P4 switches

P4 employs tables as key constructs for matching packet headers and executing corresponding actions, as depicted in Fig. 2. Each entry in a table specifies a match criterion and an associated action with parameters, allowing for dynamic adaptability to meet slicing demands. We denote $\boldsymbol{E}_{i,j}$ as the allocation of table entries for slice i in switch j, where the total capacity of each switch's table entries is C. This dynamic configuration facilitates efficient routing, bandwidth management, and QoS maintenance for each slice, ensuring optimal alignment with specific traffic profiles. We use $g(E_{i,j})$ to represent the utilization factor of the allocated table entries.

3.4 Problem formulation

Our primary objective is to optimize the alignment between user demands and actual network performance, a goal grounded in the need to maximize user satisfaction within the constraints of finite network resources. By effectively balancing these aspects, we aim to create a network slicing environment that is not only responsive to user needs but also maintains optimal operational efficiency. The utility function for meeting the requirements of user u is defined as a sum of logarithmic terms, capturing the principle of diminishing returns, which reflects the reality that incremental improvements in network performance yield progressively smaller increases in user satisfaction,

$$\text{Utility}(\boldsymbol{D}(u)) = \alpha_u \log(1+\frac{B_u^{'}}{B_u}) + \beta_u \log(1+\frac{L_{\max,u}}{L_u^{'}}) + \gamma_u \log(1+\frac{P_{\max,u}}{P_u^{'}}) \tag{1}$$

Subject to

$$\sum_{s\in S} B_s^0 \leqslant B_{\text{total}}$$

where α_u, β_u, and γ_u are weighting factors for bandwidth, latency, and packet loss, respectively, each reflecting the relative importance of these aspects in determining user satisfaction. $B_u^{'}$, $L_u^{'}$, and $P_u^{'}$ represent the actual bandwidth, latency, and packet loss experienced by the user collected from INT, while B_u,

$L_{\max,u}$, and $P_{\max,u}$ denote their respective demanded or acceptable levels. Our goal is to maximize user utility, balancing this with the cost of P4 configurations to enhance resource utilization efficiency as

$$\max \sum_{u\in U} \text{Utility}(\boldsymbol{D}(u)) - \sum_{s\in S}(\kappa_1 \times \text{CIR}_s + \kappa_2 \times \text{CBS}_s + \kappa_3 \times \text{PIR}_s) - \lambda \sum_{j\in J} g(E_{i,j}) \quad (2)$$

Subject to

$$\sum_{i\in I} E_{i,j} \leqslant C, \forall j \in J,$$
$$0 \leqslant U^t_{i,j} \leqslant 1, \forall i \in I, j \in J.$$

We denote $\kappa_1, \kappa_2, \kappa_3$, and λ as cost coefficients for CIR, CBS, PIR, and table entry utilization, respectively. The cost functions for meter configurations and table entries in the objective function can indeed be linear. This choice simplifies the formulation and aligns with the principles of mixed-integer linear programming (MILP).

4 P4-Based Fine-Grained Resources Management Method

This section presents a comprehensive examination of the programmable switch components, focusing on both the architecture and resource modeling of the programmable data plane. Our objective is to not only ensure effective bandwidth guarantees but also to allow for the interference-free sharing of residual bandwidth when design the meter in P4, as illustrated in Fig. 3.

4.1 Meter in P4

Fundamental to our strategy for P4-based bandwidth management is the implementation of the differentiated services (DiffServ) model. This approach classifies packets based on distinct code-points indicating their service level, which then governs their forwarding behavior and aligns traffic flow with predefined policy

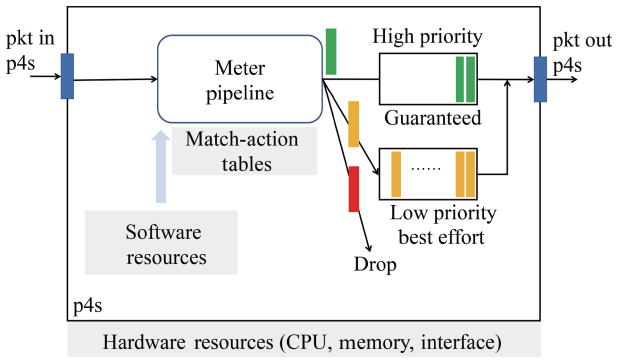

Fig. 3 Integrated view of P4 switch resources and bandwidth management.

constraints and service level agreements. Complementing the DiffServ framework, we utilize the two-rate, three-color marker (trTCM) mechanism. This mechanism segregates packets into green, yellow, or red categories, based on their ingress rates relative to the established CIR and PIR, using a dual token bucket algorithm to control the rate of transmission.

Figure 3 provides a detailed representation of a P4 switch, highlighting the interconnection between its software programming, table resources, and hardware infrastructure. The meter pipeline showcases the trTCM methodology in bandwidth management, allocating traffic to high or low priority queues based on specific criteria. Green packets, compliant with the agreed traffic profile by not exceeding the CIR, are typically given priority, supporting the network's goals of minimizing latency and maximizing throughput. Yellow packets, which exceed the CIR but remain below the PIR, indicate a tolerable deviation from the committed rate and are usually queued with intermediate priority. Red packets, exceeding the PIR, represent a significant deviation and are frequently subjected to lower priority handling or dropping, as a measure to control excessive bandwidth usage.

Algorithm 1 shows the integration of trTCM in the P4 environment enables network administrators to dynamically shape traffic and enforce policies effectively, a crucial aspect in managing the diverse demands of modern network traffic and maintaining service integrity across various network segments. Adjusting the CIR and PIR thresholds in response to real-time network conditions allows for fine-tuned service delivery, aligning with changing traffic patterns and service requirements. In order to provide a practical understanding of the metering design within our P4-based approach, we present a key segment of the P4 ingress control code. This code exemplifies how the trTCM mechanism is applied to network packets for effective bandwidth management:

4.2 A differentiated resource management approach based onmeter in P4

We aim to devise a novel network slicing solution that guarantees and manages bandwidth for each slice, leveraging the scalable resources and the flexible programmability of P4 switches. From our earlier discussions, we have established that the meter-based component, empowered by the trTCM scheme, facilitates effective bandwidth management for

Algorithm 1 P4 Ingress Control Code with trTCM settings

```
control Ingress {
apply {
if (ingress_port_acl.apply().hit) {
meter_result_t result = trTCM.execute_meter(
  packet_size, flow_index);
    switch (result.color) {
      case GREEN:
        // GREEN: For slice ID 1 with CIR of 100 Mbit/s
        if (flow_index = = 1) {
          mark_packet_dscp(EXPEDITE_FORWARDING
            );
          transmit_packet();
        }
        break;
      case YELLOW:
        // YELLOW: For slice ID 2 with CIR of 50 Mbit/s,
        enqueued to mid-priority
        if (flow_index = = 2) {
          mark_packet_dscp(ASSURED_FORWARDING)
            ;
          enqueue_packet(mid_priority_queue);
        }
        break;
      case RED:
        // RED: For slice ID 3 exceeding PIR, packets are
        dropped
        if (flow_index = = 3) {
          mark_packet_dscp(DEFAULT_FORWARDING)
            ;
          drop_packet(); // Assuming RED packets are
          dropped
        }
        break;
      }
    }
  }
}
```

different flows and efficient slice resource allocation. However, the advent of new resources necessitates more stringent traffic isolation to optimize the use of remaining bandwidth.

We have instituted a metering mechanism that employs P4's register arrays to continuously monitor the utilization of various network resources, including bandwidth, storage, and computational capacity. This dynamic surveillance permits us to modulate resource allocations based on real-time measurements. For each network slice, we configure a trTCM meter instance in P4, enforcing the committed and peak information rates (CIR and PIR).

Figure 4 illustrates the integrated resource management approach within a P4 switch, incorporating our novel design of metering combined with token buckets. This design is pivotal for fine-grained bandwidth management across network slices, as depicted by Meter_{s_1} through Meter_{s_N}, and for traffic that does not belong to a specific slice, labeled as non-sliced. Each meter is associated with a corresponding token bucket that regulates the passage of packets based on their priority level. High-priority traffic is ensured guaranteed bandwidth by its meter, ensuring compliance with predefined service-level agreements. Low-priority traffic, which does not exhaust its meter's allocated rate, is allowed to tap into unused high-priority bandwidth, thus utilizing the excess capacity in an interference-free manner. This mechanism is illustrated through two distinct pathways originating from each meter: one directs packets that meet the committed rate, encapsulated within the token bucket, to a high-priority queue for expedited processing (indicated by a green arrow representing guaranteed traffic), while the other diverts to a low-priority queue.

The token buckets serve a critical function in this architecture by dynamically adjusting to traffic conditions. They replenish their token count over time, up to the maximum burst size, allowing for short-term surges in traffic to be accommodated without penalizing the overall network performance. The token bucket associated with each meter ensures that green-coded packets are transmitted with priority, while yellow-coded packets, which exceed the committed burst size but not the peak rate, are queued for later transmission. Red-coded packets, exceeding the peak rate, are subject to possible dropping, as indicated by

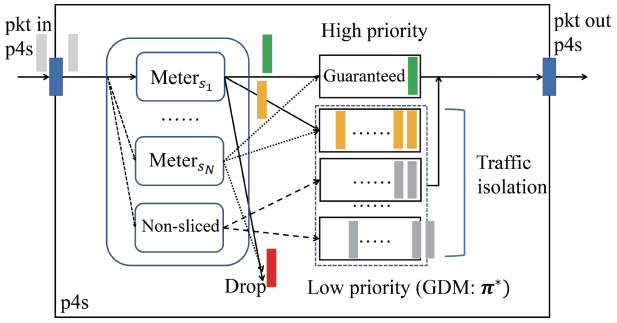

Fig. 4 Token bucket assists shared bandwidth allocation and traffic isolation.

the red "Drop" line, to prevent network congestion.

This dynamic bandwidth sharing is especially significant in scenarios where high-priority traffic does not fully utilize its reserved bandwidth. In such cases, the unused bandwidth can be temporarily allocated to low-priority traffic, enhancing overall network throughput without compromising the QoS of high-priority slices. The proposed method thus ensures that each network slice can utilize its fair share of resources, enhancing the fairness and ensuring optimal resource utilization across the network. We different slices can be managed with varying priorities within a P4 programmable switch, utilizing metering and token bucket algorithms for effective bandwidth management in Algorithm 2.

As we state the key code of P4, the set_dscp_and_transmit action within the code sets the differentiated services code point (DSCP) for the packet and designates its egress port, a crucial step in applying the QoS policies. The subsequent enqueue_yellow action showcases the token bucket mechanism in action: tokens from the committed bucket are consumed first, allowing packets that conform to the agreed-upon rate to be transmitted at a medium priority level. If the committed bucket is exhausted, tokens from the peak bucket are used, reflecting the allowance for traffic bursts. Should both buckets be depleted, packets may be dropped or enqueued at a lower priority, ensuring fair access to the network while avoiding congestion.

Algorithm 2 P4 ingress control code with trTCM utilizing metering and token bucket

```
control Ingress {
  action set_dscp_and_transmit(bit<6> dscp_value, bit<9> egress_port) {
    standard_metadata.egress_spec = egress_port;
      // Set the egress port for the packet
    hdr.ipv4.dscp = dscp_value;
  }
  action enqueue_yellow(bit<3> queue_id) {
    // The meter has a committed bucket (CB) and a peak bucket (PB)
    if (meter_cb.tokens > 0) {
      // Consume a token from the committed bucket
        meter_cb.consume_token();
      set_dscp_and_transmit(ASSURED_FORWARDING, MID_PRIORITY_EGRESS_PORT);
    } else if (meter_pb.tokens > 0) {
      // Consume a token from the peak bucket for burst allowance
      meter_pb.consume_token();
      set_dscp_and_transmit(ASSURED_FORWARDING, MID_PRIORITY_EGRESS_PORT);
    } else {
        drop_or_enqueue_packet(queue_id);
    }
  }
  apply {
    if (ingress_port_acl.apply().hit) {
      // Execute metering with token buckets to determine the packet's color
      meter_result_t result = meter.execute(packet_size, flow_index);
      // Actions are determined based on the color assigned by the meter
      switch (result.color) {
      }
    }
  }
}
```

5 Design of the Generative Diffusion Algorithm

5.1 Closed-loop of control plane and programmable data plane

As illustrated in Fig. 5, we propose an end-to-end resource management architecture for network slicing, comprising P4 switches and open-source controllers, such as ONOS. P4 switches in the architecture are configured for rapid wire-speed forwarding, crucial for maintaining the low-latency, high-throughput performance expected in 5G networks.

To complement their forwarding capabilities, the switches are outfitted with processing cards that combine computing and storage functionalities. These cards adeptly manage complex, non-time-critical processing tasks, striking an equilibrium between expeditious data handling and computational proficiency. Upon entering the system, data is initially processed by the controller and subsequently routed through the internal processing card. Here, it is directed to the application-specific integrated circuit (ASIC) based on predefined specific service (SP) conditions, which cater to specialized services or protocols. Compliant data streams follow the SP path, while others proceed along the normal service route. This bifurcation ensures that resources are judiciously

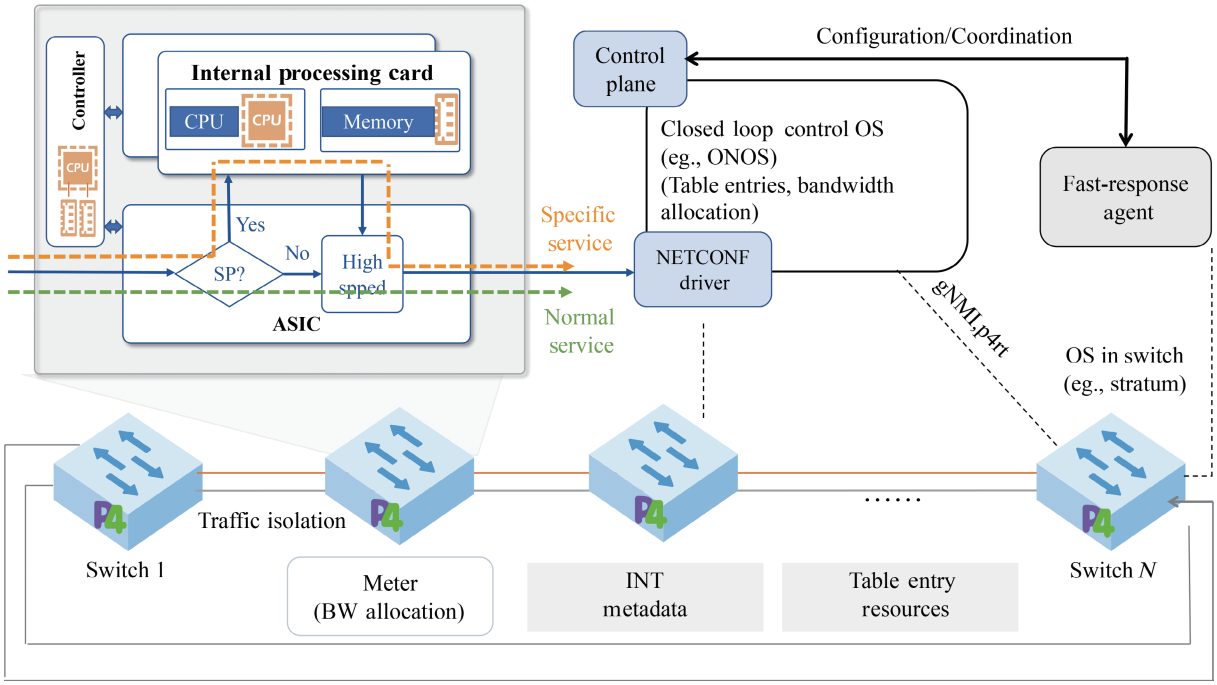

Fig. 5 An end-to-end sliced resource management architecture.

allocated and services meticulously tailored, bolstering network slicing management.

Within this framework, the control plane employs a generative diffusion model (GDM) to analyze network performance feedback from the data plane. This AI-centric method dynamically refines network slicing parameters, optimizing bandwidth management and traffic isolation based on real-time network traffic analysis and demand patterns.

5.2 Generative diffusion algorithm designed to solve bandwidth management

In our network architecture, the GDM is deployed within the control plane, applying a conditioned diffusion process to inform decision-making. This approach employs a sophisticated machine learning model, designed to generate optimal decisions based on the existing network environment, which includes incoming user tasks and the status of network resources. The primary aim of the GDM algorithm is to maximize overall utility, optimizing the allocation of key network resources, particularly bandwidth. This ensures effective traffic management and meets service requirements efficiently.

We denote the original strategy for meter as $\boldsymbol{x}_s$ for the slice s. Diffusion-based generative models frame the process of creating data as a methodical reversal of noise addition. Initially, the data is progressively obscured by noise in a forward process. These models then employ a reverse operation, iteratively refining and restoring the data's structure. This approach casts the model in the role of a latent variable system, where the data's original form is gradually uncovered through denoising steps.

$$p_\theta(\boldsymbol{x}_0) := \int p_\theta(\boldsymbol{x}_{0:T}) d\boldsymbol{x}_{1:T} \tag{3}$$

We denote the latents of the same dimensionality as $\{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_T\}$. The reverse diffusion joint distribution, $p_\theta(\boldsymbol{x}_{0:T})$, is defined as a Markov chain with learned Gaussian transitions, starting from a standard Gaussian distribution, which can be expressed as

$$p_\theta(\boldsymbol{x}_{0:T}) := p(\boldsymbol{x}_T) \prod_{t-1}^{T} p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) \tag{4}$$

where each transition is modeled by a Gaussian distribution:

$$p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) := \mathcal{N}(\boldsymbol{x}_{t-1}; \mu_\theta(\boldsymbol{x}_t, t), \sum_\theta(\boldsymbol{x}_t, t)) \tag{5}$$

Conversely, the forward diffusion chain incrementally adds noise over time through a Markov process according to a variance schedule $\{\sigma_1, \sigma_2, \ldots, \sigma_T\}$ as

$$q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0) := \prod_{t=1}^{T} q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) \tag{6}$$

where each step is described by

$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) := \mathcal{N}(\boldsymbol{x}_t; \sqrt{1-\sigma_t}\boldsymbol{x}_{t-1}, \sigma_t \boldsymbol{I}) \tag{7}$$

Following the training, the model generates data by sampling from $\boldsymbol{x}_T$ and applying the reverse diffusion chain. The model can be adapted to specific conditions by incorporating additional contextual information. Diffusion models exhibit a versatile architecture that allows for their extension to conditional models. This is achieved by incorporating conditions into the probability distribution, denoted as $p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, c)$, thereby guiding the model to generate data based on given conditions or parameters.

As shown in Fig. 6, decisions derived from the GDM in the control plane are transformed into implementable configurations and rules within the P4 programmable data plane. These decisions, formulated through the conditioned diffusion process, aim to optimize resource utilization and traffic management within the network slices. P4 switches are then programmed to enforce these strategies, encompassing traffic isolation, prioritization, and rate limiting for each slice. We depict two key step in the GDM algorithm:

(1) Forward Diffusion: In our system, network strategies for each slice begin as initial configurations, which are then subjected to a controlled forward diffusion process. This involves the systematic introduction of Gaussian noise to the contractual parameters, akin to incrementally blurring an image. This stepwise increase in noise level gradually obscures the original data, simulating the uncertainty and variability inherent in network traffic patterns.

(2) Reverse Diffusion: After the data reaches a certain noise saturation, the reverse diffusion process commences. In this phase, noise is methodically removed, or the data is denoised. This reverse process is critical: as the noise diminishes, our AI-driven algorithm, encompassing the "Strategy Quality Network" and the "Strategy Generation Network" learns to reconstruct and refine the resource allocation. This iterative learning approach guides the system towards generating strategies that progressively align closer to the ideal solution for efficient bandwidth management and resource allocation in network slicing.

The environment essentially encapsulates the static and dynamic aspects of the network that are not directly controlled by the bandwidth management strategies but instead influence or constrain them. This includes the physical network infrastructure, current network conditions (like congestion or failure states), and user requirements. We use $\boldsymbol{e}$ to represent the environment, including both static and dynamic aspects that influence or constrain bandwidth management strategies. This set of variables includes:

$$\boldsymbol{e} = \{B_u, L_u, P_u, B_i^{'}, L_i^{'}, P_i^{'}, \alpha_u, \beta_u, \gamma_u, J, C, \sigma_t\} \tag{8}$$

The AI-driven network slicing design process

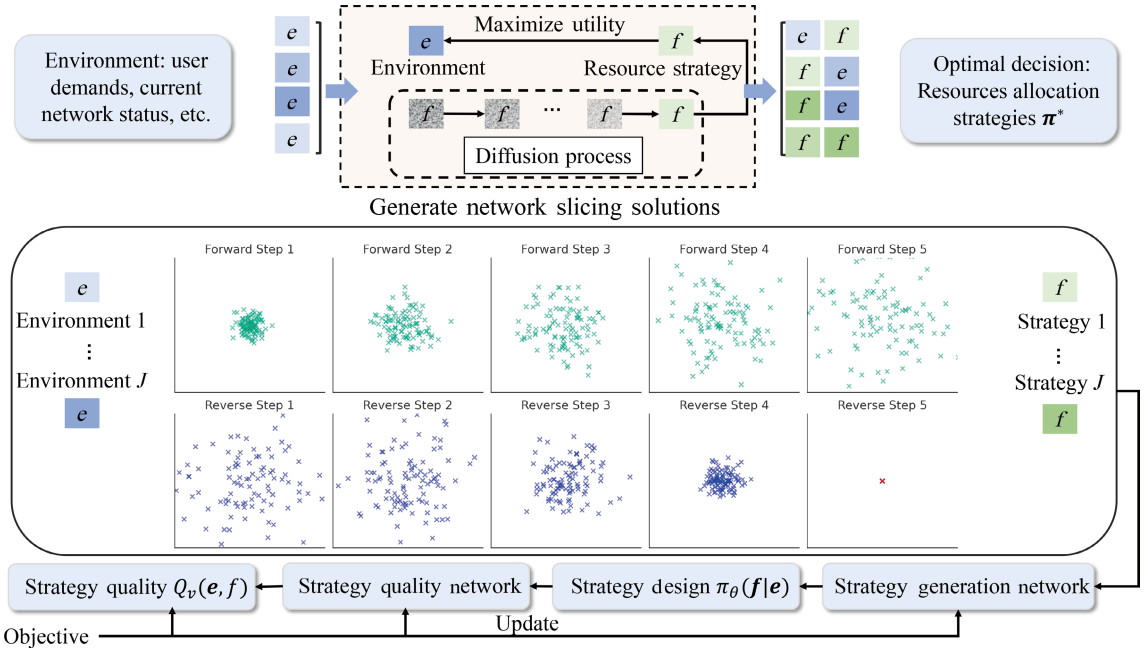

Fig. 6 Adaptive resource management in network slicing using generative diffusion models.

analyzes this environment, leveraging data from P4 meters to create and optimize network slices. We represent the bandwidth management strategies as $\boldsymbol{f}=\{\mathrm{CIR}_s,\mathrm{PIR}_s,\mathrm{CBS}_s,E_{i,j}\}$, which are adapted based on the environmental conditions. The diffusion model's policy, $\pi_\theta(\boldsymbol{f}|\boldsymbol{e})$, which maps the states of the environment to specific meter designs. This policy aims to output a deterministic meter design that maximizes expected cumulative rewards over a series of time steps, effectively translating complex environmental data into actionable network configurations. The intricate process of reverse diffusion in the conditional diffusion model is outlined as

$$\pi_\theta(\boldsymbol{f}|\boldsymbol{e})=p_\theta(\boldsymbol{f}^{0:N}|\boldsymbol{e})=\mathcal{N}(\boldsymbol{f}^N;\boldsymbol{0},\boldsymbol{I})\prod_{i=1}^{N}p_\theta(\boldsymbol{f}^{i-1}|\boldsymbol{f},\boldsymbol{e}) \quad (9)$$

This formulation clarifies the policy as a probability distribution, evolving over multiple iterations to refine the meter design towards an optimal state. As illustrated in Fig. 6, the concluding iteration of the reverse diffusion chain yields the selected meter configuration, representing the optimized design for network resource management. Here, $p_\theta(\boldsymbol{f}^{i-1}|\boldsymbol{f},\boldsymbol{e})$ can be modeled as a Gaussian distribution $\mathcal{N}(\boldsymbol{f}^{i-1};\mu_\theta(\boldsymbol{f}^i,\boldsymbol{e},i),\sum_\theta(\boldsymbol{f}^i,\boldsymbol{e},i))$. $p_\theta(\boldsymbol{f}^{i-1}|\boldsymbol{f},\boldsymbol{e})$ can be modeled as a noise prediction model with the covariance matrix fixed as

$$\sum_\theta(\boldsymbol{f}^i,\boldsymbol{e},i)=\sigma_i\boldsymbol{I} \quad (10)$$

and mean the mean is constructed to guide the reverse diffusion towards the desired meter design:

$$\mu_\theta(\boldsymbol{f}^i,\boldsymbol{e},i)=\frac{1}{\sqrt{1-\sigma_i}}\left(\boldsymbol{f}^i-\frac{\sigma_i}{\sqrt{1-\tau}}\varepsilon_\theta(\boldsymbol{f}^i,\boldsymbol{e},i)\right) \quad (11)$$

where $\tau=\prod_{s=1}^{i}a_s$ signifies the cumulative product of a sequence of parameters up to the i-th step, which influences the reverse diffusion process. To initiate the reverse diffusion, we begin by sampling $\boldsymbol{f}^N$ from a normal distribution with a mean vector of zero and a covariance matrix of the identity matrix, denoted as $\mathcal{N}(\boldsymbol{0},\boldsymbol{I})$. Subsequently, this sampled point is iteratively refined through the reverse diffusion chain, which is parameterized by θ, effectively tracing back to the optimal meter configuration.

$$\boldsymbol{f}^{i-1}|\boldsymbol{f}^i=\frac{\boldsymbol{f}^i}{\sqrt{1-\sigma_i}}-\frac{\sigma_i}{(1-\sigma_i)(1-\tau_i)}\varepsilon_\theta(\boldsymbol{f}^i,\boldsymbol{e},i)+\sqrt{\sigma_i}\varepsilon \quad (12)$$

This network is pivotal in interpreting and optimizing the meter configurations for each network slice, adapting to varying network conditions and requirements.

We introduce the slicing strategy quality network, Q_v, drawing inspiration from the Q-function used in deep reinforcement learning (DRL). This network maps each environment-meter pair, $\{\boldsymbol{e},\boldsymbol{f}\}$, to a value that quantifies the expected cumulative reward. This process of quantification is pivotal as it captures the anticipated utility from implementing a specific meter design policy. It takes into account the prevailing state of the network and projects the benefits of adhering to the chosen policy in future operations. Therefore, the ideal meter design policy, which is crucial for achieving efficient bandwidth management and effective network slicing, is determined as the one that optimizes this expected cumulative utility. This optimization ensures that network resources are allocated and managed in the most effective manner, aligning with the dynamic demands of modern network environments. Mathematically, this optimal policy can be obtained by solving the following optimization problem:

$$\pi=\underset{\pi_\theta}{\operatorname{argmin}}\mathcal{L}(\theta)=-\mathbb{E}_{\boldsymbol{f}^0\sim\pi_\theta}[Q_v(\boldsymbol{e},\boldsymbol{f}^0)] \quad (13)$$

As shown in Fig. 6, the strategy quality network, $Q_v(e,f)$, and the strategy generation network operate to evaluate and generate resource strategies. They work to map the environment-resource pairs to an expected cumulative reward value, fostering an objective-driven approach to network resource management. As the process unfolds, the algorithm converges on an optimal decision for resource allocation strategies, denoted as π^*. This strategy represents the culmination of iterative learning and refinement, aimed at achieving the best possible alignment between network conditions and resource management objectives.

6 Experiment and Analysis

For our experimental evaluation, we constructed a network within a Mininet environment, operating on a Linux 18.04 platform, that consists of two P4 switches, each establishing connections to four servers. These switches conform to the P4-16 architecture and are intricately configured to support a spectrum of up to eight priority queues. This configuration mirrors the practical bandwidth constraints of contemporary networks, with a 1 Gbit/s bandwidth per link, as

dictated by the capacity of local network interface cards.

We simulate traffic flows using iPerf, generating both UDP and TCP streams to mimic a variety of network traffic scenarios. The flows are initiated from servers attached to one P4 switch and are directed towards servers on the corresponding switch. To glean precise measurements of TCP flow round-trip times (RTT), we employ the Flowgrind tool. This tool is indispensable for our analysis, providing granular performance metrics that reveal the impact of different traffic conditions and switch configurations on network efficiency.

Adhering to the experimental parameters set forth in Ref. [10], each of our tests spans a duration of 80 seconds. The initial 19 seconds are characterized by a baseline state where no slicing is implemented, serving as a control for subsequent comparisons. Commencing from the 20th second and extending to the 39th second, we activate a network slice designed for sliced traffic flows, applying a configuration denoted as $S(0.7, 0.8)$—indicative of a 0.7 Gbit/s CIR and an 0.8 Gbps PIR. Subsequently, we introduce a variation in the slicing configuration to $S(0.2, 1)$ during the interval marked as $I[40, 59]$, simulating a dynamic adjustment of network policies. This change serves to evaluate the adaptability and responsiveness of the P4 switch under modified traffic management conditions. Finally, the system is reverted to a non-sliced state, allowing for a comparative analysis against the predefined slicing intervals. We set diffusion step $N = 10$, batch size as 512, discount factor 0.9, soft target update parameter $\tau = 0.003$, exploration noise $\varepsilon = 0.01$, the learning rate of the strategy generation network as $\varepsilon_\theta = 10^{-5}$, and learning rate of the strategy quality network Q_v as 10^{-5}.

Figure 7 presents the empirical results from our examination of traffic management via P4 switch metering capabilities, particularly focusing on the enforcement of CIR and PIR parameters across network slices. The graph delineates the throughput over time for both sliced and non-sliced UDP traffic flows, capturing the nuanced behavior enforced by the P4 switch's metering logic. Figure 7 illustrates that the sliced UDP stream consistently attains a throughput that surpasses the CIR yet remains below the PIR threshold. This phenomenon is attributed to the P4 switch's priority queueing mechanism, where the CIR bandwidth is safeguarded within high-priority queues, ensuring that the minimum service level is always met.

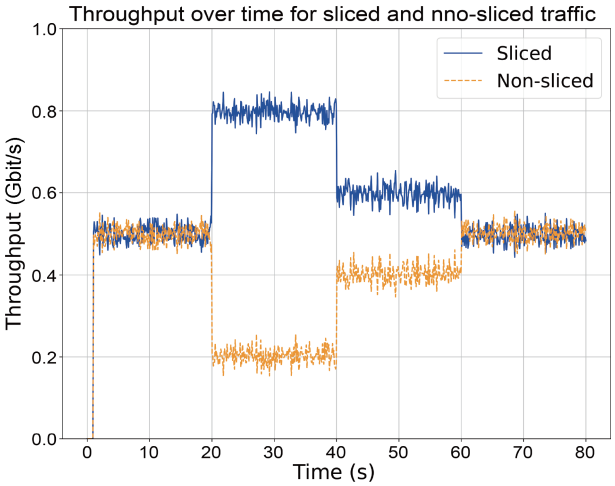

Fig. 7 Dynamic throughput allocation for sliced and non-sliced traffic in P4 meter.

Then, we evaluate the performance of the bucket for different flows. Figure 8 showcases a comparative analysis of throughput over time for various traffic flows under our P4-based network slicing strategies. This experimental insight underscores the nuanced traffic management capabilities of our P4-based slicing system, particularly the efficacy of its bucket strategy in dynamically adjusting to varying load conditions. It further validates the system's capability to maintain fidelity to slice specifications, guaranteeing service levels while ensuring fair bandwidth distribution among concurrent flows.

Then, we evaluate the performance of the generated algorithm for bandwidth allocation strategy. As depicted in Fig. 9, GDM maintains a consistently higher reward trajectory throughout the iterations,

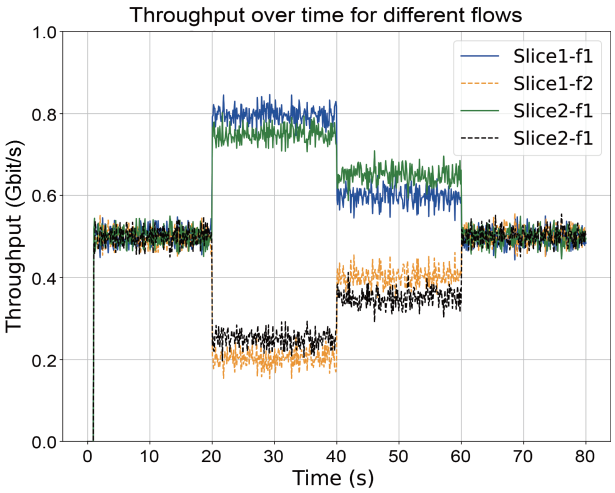

Fig. 8 Throughput evaluation of token bucket performance across different flows.

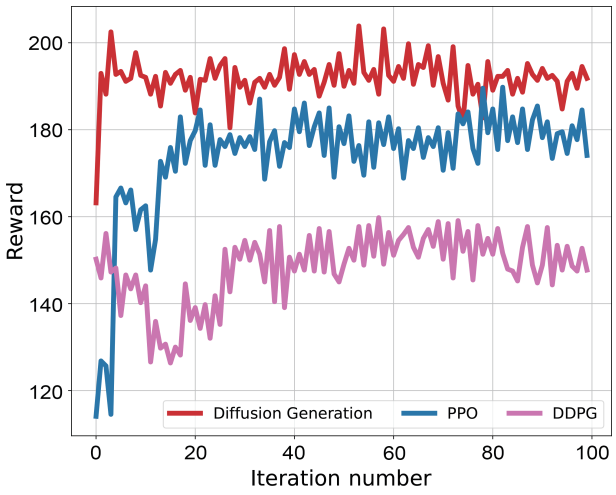

Fig. 9 Performance of GDM vs. RL algorithms.

reflecting its superior capability to converge on optimal network slicing strategies. In contrast, PPO and DDPG exhibit fluctuations in reward optimization, with DDPG demonstrating more significant variance. This variability might indicate a sensitivity to initial conditions and a longer path to convergence for traditional reinforcement learning approaches.

7 Conclusion and Future Work

In this research, we have developed an integrated framework for network slicing resource management that exploits the dynamic capabilities of P4 programmable switches and open-source controllers. The inclusion of a generative diffusion model within the control plane has significantly enhanced our system's responsiveness to the diverse requirements of 5G services. This novel GAI-centric approach allows for dynamic resource allocation that is adaptable to changing network conditions, ensuring optimal traffic differentiation and service quality. Looking forward, the framework will need to be expanded to accommodate the complexities of shared infrastructure scenarios, particularly in multi-operator hardware environments. Additionally, future developments will focus on the imperative of energy efficiency, with the aim of designing P4-based mechanisms that optimize network operations for reduced power consumption through smarter traffic shaping and resource allocation strategies, aligning network advancements with sustainable energy practices.

Acknowledgment

This work was in part supported by the funding from the National Natural Science Foundation of China (Nos. 62325203 and U22B2033), in part by the General Artificial Intelligence computing Chip project for training in 2022 (No. CEIEC-2022-ZM02-0244) from Kunlunxin (Beijing) Technology Co., LTDt, and in part by the BUPT Excellent Ph.D. Students Foundation (No. CX2023147).

References

[1] H. Zhang, N. Liu, X. Chu, K. Long, A. H. Aghvami, and V. C. M. Leung, Network slicing based 5G and future mobile networks: Mobility, resource management, and challenges, *IEEE Commun. Mag.*, vol. 55, no. 8, pp. 138–145, 2017.

[2] Y. Wu, H. N. Dai, H. Wang, Z. Xiong, and S. Guo, A survey of intelligent network slicing management for industrial IoT: Integrated approaches for smart transportation, smart energy, and smart factory, *IEEE Commun. Surv. Tutorials*, vol. 24, no. 2, pp. 1175–1211, 2022.

[3] S. Zhang, An overview of network slicing for 5G, *IEEE Wirel. Commun.*, vol. 26, no. 3, pp. 111–117, 2019.

[4] Z. Shu and T. Taleb, A novel QoS framework for network slicing in 5G and beyond networks based on SDN and NFV, *IEEE Netw.*, vol. 34, no. 3, pp. 256–263, 2020.

[5] C. Bektas, S. Monhof, F. Kurtz, and C. Wietfeld, Towards 5G: An empirical evaluation of software-defined end-to-end network slicing, in *Proc. IEEE Globecom Workshops* (*GC Wkshps*), Abu Dhabi, United Arab Emirates, 2018, pp. 1–6.

[6] P. Bosshart, D. Daly, G. Gibb, M. Izzard, N. McKeown, J. Rexford, C. Schlesinger, D. Talayco, A. Vahdat, G. Varghese, et al., *SIGCOMM Comput. Commun. Rev.*, vol. 44, no. 3, pp. 87–95, 2014.

[7] Y. W. Chen, L. H. Yen, W. C. Wang, C. A. Chuang, Y. S. Liu, and C. C. Tseng, P4-enabled bandwidth management, in *Proc. 2019 20th Asia-Pacific Network Operations and Management Symposium* (*APNOMS*), Matsue, Japan, 2019, pp. 1–5.

[8] E. Hauser, M. Simon, H. Stubbe, S. Gallenmüller, and G. Carle, Slicing networks with P4 hardware and software targets, in *Proc. ACM SIGCOMM Workshop on 5G and Beyond Network Measurements, Modeling, and Use Cases*, Amsterdam, the Netherlands, 2022, pp. 36–42.

[9] S. Y. Wang, H. W. Hu, and Y. B. Lin, Design and implementation of TCP-friendly meters in P4 switches, *IEEE/ACM Trans. Netw.*, vol. 28, no. 4, pp. 1885–1898, 2020.

[10] Y. W. Chen, C. Y. Li, C. C. Tseng, and M. Z. Hu, P4-TINS: P4-driven traffic isolation for network slicing with bandwidth guarantee and management, *IEEE Trans. Netw. Serv. Manag.*, vol. 19, no. 3, pp. 3290–3303, 2022.

[11] Z. Chen, Z. Zhang, and Z. Yang, Big AI models for 6g wireless networks: Opportunities, challenges, and research directions, arXiv preprint arXiv: 2308.06250, 2023.

[12] J. Ho, A. Jain, and P. Abbeel, Denoising diffusion probabilistic models, *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.

[13] H. Du, R. Zhang, Y. Liu, J. Wang, Y. Lin, Z. Li, D. Niyato, J. Kang, Z. Xiong, S. Cui, et al., Beyond deep reinforcement learning: A tutorial on generative diffusion models in network optimization, arXiv preprint arXiv: 2308.05384, 2023.

[14] D. Bega, M. Gramaglia, M. Fiore, A. Banchs, and X. Costa-Perez, DeepCog: Cognitive network management in sliced 5G networks with deep learning, in *Proc. IEEE INFOCOM 2019 - IEEE Conf. Computer Communications*, Paris, France, 2019, pp. 280–288.

[15] S. Jošilo and G. Dán, Joint wireless and edge computing resource management with dynamic network slice selection, *IEEE/ACM Trans. Netw.*, vol. 30, no. 4, pp. 1865–1878.

[16] A. Thantharate and C. Beard, ADAPTIVE6G: Adaptive resource management for network slicing architectures in current 5G and future 6G systems, *J. Netw. Syst. Manag.*, vol. 31, no. 1, pp. 9, 2022.

[17] T. Mai, H. Yao, N. Zhang, W. He, D. Guo, and M. Guizani, Transfer reinforcement learning aided distributed network slicing optimization in industrial IoT, *IEEE Trans. Ind. Inf.*, vol. 18, no. 6, pp. 4308–4316, 2022.

[18] R. P. Pinto, K. S. Mayer, D. S. Arantes, D. A. A. Mello, and C. E. Rothenberg, Packet-optical differentiated survivability implemented by P4 slices and gNMI telemetry, in *Proc. Optical Fiber Communication Conf. (OFC) 2023*, San Diego, CA, USA: Optica Publishing Group, 2023, pp. M1G–3.

[19] M. Xu, H. Du, D. Niyato, J. Kang, Z. Xiong, S. Mao, Z. Han, A. Jamalipour, D. I. Kim, V. Leung, et al., Unleashing the power of edge-cloud generative AI in mobile networks: A survey of aigc services, arXiv preprint arXiv: 2303.16129, 2023.

[20] Y. Huang, M. Xu, X. Zhang, D. Niyato, Z. Xiong, S. Wang, and T. Huang, Ai-generated 6g internet design: A diffusion model-based learning approach, arXiv preprint arXiv: 2303.13869, 2023.

[21] X. Huang, P. Li, H. Du, J. Kang, D. Niyato, D. I. Kim, and Y. Wu, Federated learning-empowered AI-generated content in wireless networks, arXiv preprint arXiv: 2307.07146, 2023.

[22] H. Du, R. Zhang, D. Niyato, J. Kang, Z. Xiong, D. I. Kim, X. S. Shen, and H. V. Poor, Exploring collaborative distributed diffusion-based AI-generated content (AIGC) in wireless networks, *IEEE Netw.*, pp. 1–8, 2024.

Haipeng Yao is a professor in Beijing University of Posts and Telecommunications. He received the PhD from Beijing University of Posts and Telecommunications in 2011. His research interests include future network architecture, network artificial intelligence, networking, space-terrestrial integrated network, network resource allocation, and dedicated networks. He has published more than 150 papers in prestigious peer-reviewed journals and conferences. Dr. Yao has served as an associate editor of *IEEE Transactions on Mobile Computing*, and *IEEE Transactions on Sustainable Computing*. He has also served as a member of Technical Program Committee as well as Symposium Chair for a number of international conferences, including IWCMC 2019 Symposium Chair, and ACM TUR-C SIGSAC2020 Publication Chair.

Huan Chang is an assistant professor with School of Information and Electronics, Beijing Institute of Technology (BIT). She received the PhD degree from Beijing University of Posts and Telecommunications (BUPT), China, in 2020. Her main research interests include large-capacity optical communication and adaptive optics.

Wenji He received the bachelor degree from Beijing University of Posts and Telecommunications in 2021. She is pursuing the PhD degree at School of Information and Communication Engineering, Beijing University of Posts and Telecommunications. Her research interests are in the areas of intelligent network and the programmable network.

Yunjie Liu received the BS degree in technical physics from Peking University, Beijing, China, in 1968. He is currently the academician of China Academy of Engineering, the chief of the Science and Technology Committee of China Unicom, and the dean of School of Information and Communication Engineering, BUPT. His research interests include next generation networks and network architecture and management.