Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 10, No. 3, October : 2023

SCRUTINY SCHEDULED IN CONTRAPTION LEARNEDNESS ENABLED COMPLEX - SLICING: SHEATHING THE JAM-PACKED PRESENCE SEQUENCE

Dr. Saravanan Suba , Assistant Professor, Department of Computer Science, Government Arts and Science College, Srivilliputtur - 626 125. Email-ID: drsaravanansuba@gmail.com
Dr. Dinesh Senduraja Ph.D. Research Associate (RA) , MED & COS, Defence Research & Development Organisation (DRDO) Pune- 411 021 **&** Lecturer, Department of Computer Science Government Art and Science College, Veerapandi, Theni -625 534 neo12max@gmail.com

Abstract—
Complex Slicing (CS) is becoming an necessary constituent of tune administration and orchestration in communication complex, starting from mobile cellular complex and extend to a universal proposal. CS can reshape the operation and operation of traditional tunes, support the prologue of new ones, vastly advance how resource allocation performs in complex, and notably change the user experience. Most of these promises still need to reach the real world, but they have previously established their potential in many untried infrastructures. However, difficulty, scale, and vitality are pressuring for a Engine learning (EL)-enabled CS move toward in which autonomy and competence are critical features. This trend is comparatively new but growing fast and attracting much attention. This article scrutiny Artificial Intelligence-enabled CS and its potential use in current and future infrastructures. We have covered state-of-the-art EL-enabled CS for all complex segments and planned the symbols according to the phases of the CS being series. We also discuss challenges and opportunity in research on this topic.

Index Terms—Complex Slicing, EL-enabled slicing, Engine Erudition, Slicing-as-a-Tune, EL-enabled supply Orchestration, and Allocation.

INTRODUCTION

Over the last decade, wireless complex technology has been chiefly driven by advanced complex applications such as Industry 4.0, immersive media applications (e.g., virtual/augmented/mixed reality), and mission-critical tunes (e.g., self-driving vehicles and automated traffic control sys- [1]. Following this trend, the aft-generation (5G) cellular complex have been designed to provide higher latency, bit rate, and reliability performance, fostering the digital trans- formation of vertical industries [2]. A requirement to achieve this aim is to support different communication tunes, e.g., Engine Type Communication (match), Enhanced Mobile Broadband (ebb), ultra-Reliable and Low-Latency Communications (URLLC), with highly different needs, over a shared complex infrastructure [3]. To address this challenge, 5G and beyond 5G complex embrace the concept of Complex Slicing (NS) [4]–[6], which logically divides the operator's complex into isolated, tune-tailored, end-to-end complex referred to as complex slices. The NS concept brings several advantages to complex operators [7]. First, NS allows numerous tenants to share the same physical complex infrastructure and reduce complex deployment and operation costs. Second, with NS, each complex slice is instantiated to satisfy a fraction collar set of applications, enable tune differentiation and guaranteeing Tune Level Agreement (SLA) for each application type. Finally, NS increases edibility in complex running, as complex slices can be created, muddied, and decommissioned as needed. However, to fully exploit the advantages of NS, operators have to provide dynamic resource allocation, tune assurance, isolation and protection, and optimized fraction toning of resources across all complex do chiefs, i.e., Broadcasting contact Complex (RAN), Transport Complex (TN), and Core Complex (CC), and throughout the full slice being series, from the slice preparation to the slice decommissioning. Therefore, the banes of NS come at the price of higher complexity in operating and managing wireless complex.

Currently, the realization of the NS concept relies a lot on paradigms such as Complex

Function Virtualization (CFV), Software-Dined Complex (SDN), and blurs computing. Together, these technologies provide the means of control for dynamically allocate the necessary resource capacities across the complex and resizing and moving workloads at runtime to meet the needs of tunes, regardless of complex conditions [14]. However, although these means of control are already available, the decision-making process that triggers their execution depends on static policies and human intervention [1], [22]. Therefore, the full realization of the NS paradigm depends on further automation and the closure of running control loops.

Due to advances in algorithms and the increase in computational power, in recent years, Artificial Intelligence (AI), and Engine Erudition (ML) in fraction ocular, has become an essential enabling technology to achieve good performance in complex decision-making difficulty [23]. Indeed, ML techniques are enablers of numerous difficulty involving multiple objectives subject to many heterogeneous and dynamic requirements [7], [24]. NS, in turn, is a current trend that, as a difficulty, inherently has multiple objectives, potentially deals with many do chiefs and technologies, and supports numerous users and heterogeneous requirements. Therefore several works have applied ML to deal with distinct challenges during the slice being series. Yang et al. [25] proposed an intent-driven optical NS that maps high-level intents into slice requirements for the transport complex using Latent Dirichlet Allocation. Sciancale- pore et al. [26] designed an online complex slice broker that decides which slices to accept while opportunistically pursuing

TABLE I
COMPARISON OF RELATED SCRUTINY.

Paper	Chief focus	Focus on ML	Existing works are scrutinized	Study’s Orientation
[6]	NS concepts	C	√	Background-oriented
[8]	NS concepts	C	√	Background-oriented
[9]	NS concepts	C	√	Background-oriented
[10]	NS concepts	C	C	Background-oriented
[11]	NS concepts	C	√	Background-oriented
[12]	NS implementation aspects	C	√	SDN and NFV-oriented
[13]	NS implementation aspects	C	√	SDO-oriented
[14]	NS implementation aspects	C	√	Complex segment-oriented
[15]	NS implementation aspects	C	√	IoT-oriented
[16]	NS algorithmic aspects	C	C	VNF placement-oriented
[17]	NS algorithmic aspects	C	√	Resource allocation perspective
[18]	NS algorithmic aspects	C	√	Resource allocation perspective
[19]	NS algorithmic aspects	√	√	Resource allocation perspective
[7]	NS algorithmic aspects	√	√	RAN-oriented
[20]	NS algorithmic aspects	√	C	LCM-oriented
[21]	NS algorithmic aspects	√	C	LCM-oriented

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 10, No. 3, October : 2023

This scrutiny	NS algorithmic aspects	✓	✓	LCM-oriented

the NS multiplexing gain maximization using a variant of the Multi-Armed Bandit (MAB) model. Kabila et al. [27] formulate the multi-do chief slicing as a multisubstrate Virtual Complex Embedding (VNE) difficulty and proposed a Deep Reinforcement Erudition (DRL) algorithm to solve it. Began et al. [28] proposed a Deep Erudition (DL) algorithm that anticipates future slice needs and timely reallocates/deal locates resources where and when they are required. Although these works have shown the potential of ML for supporting the budding need for autonomous complex slice operation and running, the literature has only unsystematically ad- dressed individual difficulty. as a result, there is a need to investigate and reorganize the current proposals for a comprehensive view of the primary complex slice Being Series Running (LCM) difficulty and the existing ML proposals to deal with them.

interrelated scrutiny

Several existing scrutiny have discussed the implications of the NS concept for next-generation mobile complex. Fouke's et al. [6], Afolabi et al. [8], Kaloxylos [9], Zhang [10], and Khan et al. [11] provided the research community with a general understanding of the topic, addressing NS in terms of basic concepts, enabling technologies, use cases, and challenges.

Some scrutiny have discussed the implementation aspects of NS. Barakabitze et al. [12] provided a comprehensive review of solutions for NS using SDN and NFV. Various 5G architectural approaches were compared in terms of practical implementations in their work. Chamber et al. [13] and Ordonez-Lucama et al. [14] focused on the ongoing work on NS modeling in RAN, TN, and CN do chiefs performed by different Standards Developing Organization (SDO). Wijethi- lama and Liyanage [15] studied the contribution of NS to the Internet of Things (Iota) realization.

The algorithmic aspects of NS have also been discussed in the literature [16]–[18]. Specifically, Vassilaras et al. [16] formulated NS as an optimization difficulty of placing Ver.- utilized Complex Functions (VNFs) over a set of applicant
locations and deciding their interconnections. Su et al. [17] scrutiny al the resource allocation schemes for NS using three mathematical models: game theory, forecast techniques, and sturdiness/malfunction recovery models. Dubai et al. [18] reviewed the state-of-the-art NS concerning two algorithmic challenges: slice resource allocation and slice orchestration. Nevertheless, these scrutiny considered only a few algorithmic aspects of NS, and none focused on ML solutions. Indeed, the need to use ML for complex slice operation and management was rest discussed by Kale et al. [20]. The authors describe the organization functions of complex slices that could be automated using ML and listed relevant techniques for automating such functions. However, the authors did not scrutiny existing works and proposed solutions. More recently, Sheen et al. [7] scrutiny ML solutions applied to intelligent NS running. Nevertheless, the authors considered only three septic RAN difficulty: edible broadcasting contact NS, automatic Broadcasting Contact Technology (RAT) selection, and mobile edge caching and content delivery. Wu et al. [21] discussed a broad picture of the role of AI in sixth-generation (6G) complex, highlighting potential NS difficulty where AI could be applied to facilitate bright complex running. However, similar to [20], the authors did not scrutiny existing works and proposed solutions. Ssengonzi et al. [19] presented a scrutiny of 5G NS and virtualization from a Reinforcement Erudition (RL) and DRL perspective. Nevertheless, the authors focused only on existing RL and DRL approaches and a few NS difficulty, such as resource allocation, admission control, and traffic forecasting.

Table I summarizes the chief characteristics of existing scrutiny and our work, comparing them in terms of their chief focus (i.e., NS concepts, NS implementation aspects, or NS algorithmic aspects), whether ML is measured, whether existing solution are discussed, and the key criteria

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 10, No. 3, October : 2023

driving the study. As illustrated in the table, a comprehensive scrutiny of ML applied to solve complex slice LCM difficulty is still mislaid
investigate extent and method

The chief aim of this work is to supply the booklover with a comprehensive scrutiny of the use of ML for intelligent complex slice LCM, from the slice grounding to their de- commissioning, after the 3rd Generation Fraction reship Project (3GPP) being series [29] and jacket all complex do chiefs (RAN, TN, and CN). We studied and assessed high-quality research published since 2016, available in the vehicles In- statute of Electrical and Electronics Engineers (IEEE)Explore, Association for figure Enginery (ACM) Digital Library, Science Direct, and Wiley Online Library. We introduce the existing works in terms of the intricacy they address (e.g., slice access control, keep portion, VNF assignment) after the 3GPP slice being series. Fig. 1 illustrate the union of the editorial while Table V summarize the generally-used abbreviation.

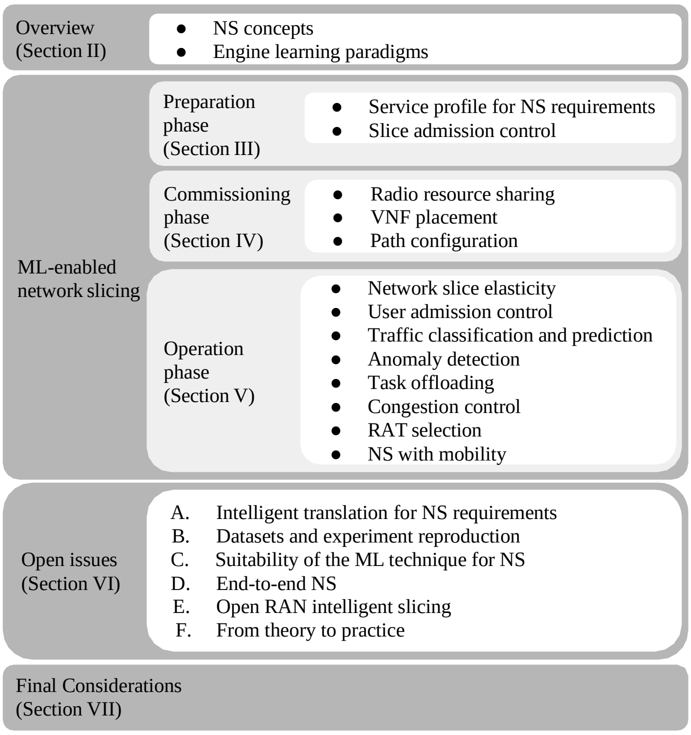

Fig. 1. The structure of the scrutiny.

At the beginning of the slice being series, i.e., in the preparation phase, ML is chiefly employed to translate tune prowls into slice requirements and to provide a slice ad- mission control. In the commission phase, ML is applied for slice resource allocation, slice VNF placement, and slice path configuration. Next, when the slice becomes operational, ML is employed for numerous runtime tasks, including user admission control, task on loading, slice elasticity, anomaly detection, RAT selection, traffics classification and prediction, congestion control, and mobility running. We point out that our scrutiny did not and works applying ML to difficulty related to slicing termination, i.e., to the decommissioning phase. Therefore, decommissioning is not illustrated in Fig. 1. In addition, our scrutiny classiest each article according to the chief difficulty it addresses, even when the article focuses on multiple difficulty and being series phases. For example, some research efforts describe using AI in two or more being series phases, such as [30]. We classify such works according to their chief addressed difficulty. Finally, although some works have proposed solutions to complex slice LCM difficulty using heuristic and genetic algorithms, our scrutiny focuses on supervise, unsupervised, reinforcement, and emerging erudition paradigms. In summing up, our chief contribution as a scrutiny is to bring a big picture of the state-of-the-art ML-enabled NS and organize the existing works from the complex slice being series outlook, illustrating the AI/ML methods used in the separate phases of the complex slice being series. then, we carefully examine every editorial and choose the being sequence phase it its based on the complexity addressed.

This article is organized as follows. Section II presents an overview of the chief concepts related to NS, focusing on NS running. Next, we thoroughly review the state-of-the-art solutions for intelligent NS running. We split the related discussion into ML-enable solutions for NS difficulty during the preparation phase (Section III), the commissioning phase (Section IV), and the operation phase (Section V). We discuss some open research issues and summarize potential future directions in Section VI. Finally, Section VII concludes the article.

OVERVIEW AND BACKGROUND

This section focuses on background concepts and key entities related to NS implementation and slice LCM to get deeper insights into the NS being series difficulty. Following the nomenclature proposed by 3GPP, a complex slice, or slice, is a logical complex comprise one or more tune chains formed by virtualized or physical complex functions and the (physical/virtual) links connecting them. This logical complex is created with appropriate isolation, resources, and optimized topology to serve one or more communication tunes [29]. communiqué tune is the term used to refer to the tenant-ordered tune. Usually, the announcement tune is expressed by a tune parole comprising the tune type, and a tune graph, where nodes represent computing/storage resources and tune instances and edges denote constraints on link bandwidth or packet loss. A complex slice can host several communication tunes if they do not impose convicting requirements.

A complex slice is an end-to-end concept, i.e., the logical complex can span across all the technical do chiefs (or segments) within the operator's complex, including the RAN, TN, and CN do chiefs. In the 5G architecture, the RAN do chief connects User Equipment (UE) to the operator's complex using various phone technologies. The TN do chief provides infrastructure connectivity between the RAN and the data complex using any technology (Internet Protocol (IP), optical, microwave, or other technology), tunnel (IP/Multiprotocol Label Switching (MPLS)), and layer functions [13]. Finally, the CN do chief allows UE to send/receive data to/from the datacomplex, providing signaling procedures such as association, register, mobility management, and meeting management
The fraction of the slice spanning a technical do chief is called a Complex Slice Subnet (NSS). Each NSS is usually deployedas a set of complex functions.

Since a complex slice can host multiple tunes, its being series within the operator's complex is independent of its associated tune(s) being series. In fraction ocular, the being series of a complex slice has four chief phases: Preparation, Commissioning, Operation, and Decommissioning. In the preparation phase, the slice does not exist. Indeed, the preparation phase comprises all preparation steps that precede slice instantiation, such as slice design, slice on boarding, and slice admission control. In the rest step, the tenant dense the tune parole from which the slice requirements will be derived. In the second step, the tenant uploads the VNFs that constitute the slice to the operator's classification The last step in the prep- ration phase decides whether the tenant NS request should be accepted or rejected based on current system utilization. In the commissioning phase, resources are assigned to the admitted slice request. Therefore, the slice is instantiated, congaed, and activated over the operator's infrastructure according to its requirements. In the action phase, the slice instance goes into operation, and its behavior is monitored to ensure compliance with the denned requirements. In this phase, runtime tasks such as upgrade, re configuration, scaling, and capacity changes can be carried out to modify the slice instance and ensure that it is optimized for its purpose. Finally, the slice instance is terminated in the decommissioning phase, and its allocated resources are released.

Since our focus is on AI/ML solutions to NS difficulty, it is imperative to introduce the ML paradigms, which are trade- tonally classier into three types: Supervised Erudition (SL), Unsupervised Erudition (UL), and RL. SL uses labeled training datasets to build models and is usually employed to solve classification and regression difficulty to predict outcomes. UL creates models using unlabeled training datasets, chiefly employed for clustering difficulty. In RL, an agent interacts with the environment via perception and action to learn areward or utility. Therefore, an RL agent learns by exploring the setting instead of being taught by exemplars. The literature has applied the aforementioned paradigms to solve some of the NS difficulty we envelop in this scrutiny. In addition, emergent learning paradigms such as Federated Erudition (FL) and Transfer Erudition (TL) have also been employed in some works. FL focuses on decentralization erudition, where distributed servers train models with local data. TL aims to utilize the built facts of a certain system to solve a different but related difficulty. We refer readers unfamiliar with these paradigms to an opening in [24], [31], [32].

ML FOR NS IN THE PREPARATION PHASE

Condition-of-the-art NS solution applied ML techniques for two difficulties in the grounding phase. First, we discuss the translation of tune prowls into slice necessities. Afterward, we present the slice admittance manage

Translation of Tune Prowls into NS Requirements

From the tenant's perspective, conniving a complex slice is a complex task that involve a complete account of the melody topology, details on tune conga oration and work bows, and SLA dentitions for tune assurance. To make this task easier, complex operators provide generic slice templates to be used as a orientation by the tenants when ordering a complex slice. However, some tunes may not have a direct mapping to a pre denned slice template since tune requirements may vary widely. For instance, some tunes may have ultra-low latency, high bandwidth, and high-reliability requirements at the same time. An alternative to this difficulty is to derive the slice requirements from tune prowls denned through high-level intents. even with much work on intent-driven compound [33], [34], we found only two articles addressing the intent-based design of complex slices.

The rest work, future by Gristly et al. [35], takes into consideration the set of tenant's intents, expressed as Quality of Tune (Quos) requirements, and the operator's grouping policies dining the supported slice types and their Quos characteristics. The aim is to determine all slice solutions underneath the tenant's order compliant with the operator's grouping policies. To this end, the approach rest maps the slice type(s) to each intent, mapping them separately to the operator's policies. It then merges these slices based on criteria such as the operator's policy they comply with and isolation and placement constraints. However, the approach accessible in [35] is model-based and, thus, does not use ML. The second work, proposed by Yang et al. [25], develops a mechanism based on ML to translate tune intents into a slicing configuration language. The predictable instrument employs the Latent Dirichlet Allocation algorithm to extract keywords from an optical complex topic model and construct an intent theme model. The intent issued by the users is a mixed distribution of certain topics, which is also a likelihoodsharing of words. If the intent topic is found, the keyword in the topic is also the core meaning of this intent. To associate intent keywords with Quos constraints, the authors propose using an experienced database. The assessment uses a discrete intent tune emulator and a complex topology assembled by Open AI Gym.

Slice Admission Control

Over-provisioning is not possible in 5G and past 5G since infrastructure resources (especially spectrum) are imperfect. Therefore, complex operators must decide which slice requests should be admitted or rejected in the infrastructure to manage resources efficiently. Specifically, the slice admission manage difficulty is formulated as follows. Upon receiving a complex slice request from a tenant, the operator's system must choose whether to accept or reject the tenant's request, pursuing a preened objective while still honoring the agreed SLAs for previously accepted complex slice requests. Such a decision is challenging as it must believe the total available system capacity, randomly arriving tenant requests, real net- work operation within the already instantiated slices, and the Quality of Experience (QoE) professed by the end-users. This section introduces recent works that applied ML to solve the slice admittance control complexity.

Began et al. [36] address the difficulty of designing a slice ad-mission control that maximizes the complex worker proceeds while pleasing the desired tune guarantee. The authors consider two types of slices: elastic, which does not require any immediate throughput guarantees, and inelastic, which requires a sure axed throughput to be satisfied during the full slice organism series. Only the RAN section is careful in work. The slice type, slice period, the slice size in conditions of the figure of users, and the value per time unit typify a slice request. The complexity is formulated as a Semi-Markov Decision Process (SMDP), where the elastic and inelastic complex slice requests follow a Poisson process. Each state is modeled as a three-sized tulle in lieu of the number of elastic and inelastic slices in the system at a given decision time and the next event (new arrival of an elastic or inelastic slice request or de fractionate of a slice of any type) that triggers a decision process. The possible actions include admitting a new request for an elastic or inelastic slice or rejecting the new request. In the rest case, the resource associated with the request is decided to the tenant, and the worker immediately earns a reward, compute as the product of the slice type price and length. The second case has no immediate reward, but the resources re chief contactable for future requests. Requests that are surplus are no longer considered by the system. The SMDP difficulty is then solved using Q-Erudition (QL), an RL algorithm where the learning function that maps the input state to the expected reward when taking a septic action is realized as a investigate table. Simulations with the slice period following an exponential sharing showed that QL achieved close to optimal performance.

Despite the good performance, an inherent drawback of RL algorithms such as QL is their lack of scalability when the state space becomes too large. Inspired by this limitation, in a later work, Began et al. [37] propose a Deep Q-Erudition (DQL) algorithm, named NS Neural Complex Admission Control (N3AC), to solve the slice permission difficulty. DRL algorithms use Neural Complex (NNs) to generalize the knowledge learned from some states to be applied to other states with similar features. In fraction ocular, N3AC uses a feed- forward NN structure, where the neurons of one layer are fully interconnected with the neurons of the next. In addition, N3AC relies on a single hidden layer and uses the Gradient Descent approach to back-propagate the measured error at the output layer to the input layer. Furthermore, N3AC does not apply any ground truth to train the NN. This training is achieved using output estimations, which become more accurate as explorations are performed. The performance of N3AC was evaluated through simulation where the tune time follows an exponential distribution and slice request arrivals follow a Poisson process.

Similar to [37], Bari et al. [38] proposed a DQL algorithm to solve the slice admission difficulty. The authors compare the performance of the DQL solution with two other algorithms: QL, and be sorry Matching. The QL and DQL approach are evaluated using the offline account of the algorithms, while Regret Matching performs online. Results show that Regret identical reacts faster to load change than the other two algorithms.

Dandachi et al. [39] propose a slice entrance control con- side ring communication, computing, and storage space resources to maximize resource utilization and operator revenue. The slice admission control considers two types of slices, Best Effort (BE) and guaranteed Quos slices, with elastic requirements. Resources from the RAN and CN do chiefs are considered. The slice admission control comprises two steps: at the commencement of each time slot, the slice admission control evaluates the similarity between the income requests and the slices already active in the system to recognize slice instances that can serve the new slice requests with a minimum amount of additional resources. The rest step uses a normalized spectral clustering algorithm based on the Jacquard similarity, while the second is implement using State-Action-Reward-State-Action (SARSA). In the second step, based on the current state of system utilization, the admission control rest decides whether to scale down the resources allocated to BE slice instances, then selects the income slice requests to admit. estimate is carried out by simulation using slice templates customized by the

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 10, No. 3, October : 2023

authors.

Reza et al. [40] propose an RL agent to decide whether or not a new slice request should be accepted. A slice request is specie in terms of its duration, tune type (priority), and the number of Central Processing Unit (CPU) and link resources needed. The purpose is to maximize the complex operator's total revenue while matching the melody requirements of the slices in operation as closely as possible. The work focuses on the RAN do chief. The RL agent is implemented using a NN that receives the slice request and the resources currently available in the system as input. NN minimizes the loss of revenue resultant by rejecting the slice requests and the loss derived by degrading the tune of a slice in operation. NN is trained in an episodic manner, and at the end of an episode, the increasing reward for all the actions up to the current point in time is compute. Evaluation is performed using a custom-built simulator, where the inter- arrival time of requests and slice duration are exponentially distributed.

Akashi et al. [41] propose a slice admission control in a federated environment formed by one consumer and provider d o chief. For a given slice request, the admission control decides whether to deploy the slice in the consumer or the provider do chief or reject it. The decision is based on the cost of deploying the slice locally (consumer do chief) or remotely (provider do chief) and on the current resource availability. The model is focused on computing resources, thus more suitable for the edge, CN, and cloud do chiefs. The authors compare the performance of two RL algorithms for solving the difficulty: QL and R-Erudition, an average reward erudition algorithm. Results are obtain through simulation using customized slice templates and show that R-Erudition performs better than QL for the federated difficulty due to QL's dependency on the discount factor.

Sciancalepore et al. [26] propose the concept of slice overbooking, where more slice requests are admitted than the overall system capacity to maximize the operator revenue. In their proposal, a slice command comprises the amount of physical wireless resources assigned to the slice and its duration. The slice entrance control difficulty is formulate as an online close process using a variant of the MAB model. Each tenant

TABLE II
SUMMARY OF ML-APPROACHES FOR NS DIFFICULTY IN THE PREPARATION PHASE.

Ref.	NS Difficulty	Erudition Paradigm	Erudition Method	Resource Type	Complex Segment	Performance Evaluation
[25]	Tune profile translation	NLP	Latent Dirichlet Allocation	Complex, computing, storage	TN	Emulation - Custom-built intent tune
[36]	Slice admission control	RL	QL	Complex	RAN	Simulation - Poisson distribution for request and exponential distribution for slice duration
[37]	Slice admission control	RL	DQL	Complex	RAN	Simulation - Similar to [36]
[38]	Slice admission control	RL	QL, DQL, Regret Matching	Complex	RAN	Simulation - Similar to [36]
[39]	Slice admission	UL, RL	Norm. spectral clustering,	Complex, computing,	RAN, CN	Simulation - Customized slice

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 10, No. 3, October : 2023

	control		SARSA	storage		templates
[40]	Slice admission control	RL	DRL	Complex, computing	RAN	Simulation - Exponential distribution for request inter-arrival time and slice duration
[41]	Slice admission control	RL	QL, R-Erudition	Computing	Edge, CN, cloud	Simulation - Customized slice templates
[26]	Slice admission control	RL	UCB, ONETS, ϵ -Greedy	Complex	RAN	Simulation and experimental

is a bandit that, if pulled at a certain round, returns a fraction ocular reward. Multiple bandits can be pulled at a given round, and tenants with active slices must be selected while their slices are operational. If a lock-up period runs, the gambler must select the same arm as in the earlier round. The reward accounts for the total amount of resources asked within the slice request and the ratio flanked by what has been used and what is being asked, following the rationale that tenants under-utilizing assigned resources are preferred for the ones fully using them. The authors realize the MAB model using three RL algorithms: Upper Condense Bound (UCB), Online Complex Slice Broker (ONETS), and ϵ -Greedy, providing a trade-off between difficulty and sub-optimality. act evaluation is accepted out by imitation. In addition, proof-of- concept implementation is accessible considering three complex slices: ebb for Guaranteed Bit Rate, ebb for BE, and Public Safety. Table II summarizes the chief individuality of the literature related to ML applied to NS difficulty in the preparation phase.

ML FOR NS IN THE COMMISSIONING PHASE

In the commission phase, NS difficulty are essentially related to making resource allocation decisions for the ad- mitted slices. After being admitted to the system, the slice is instantiated by allocating resources in the RAN, TN, and CN do chiefs. A RAN slice subnet comprises the broadcasting contact and processing function from a set of Base Stations (BSs) and the billed Physical supply Blocks (PRBs) to support a communiqué tune. A CN slice subnet contains a set of compound tunes functionalities and associated computing resources. A TN slice subnet, on the other hand, comprise a set of relatives flanked by a group of virtual or/and physical complex functions from both the RAN and the CN, each one having its own SLA. This section discusses state-of-the-art ML solutions for instantiating a slice within the RAN, TN, and CN do chiefs. First, we argue ML resource portion solution for instantiating a RAN slice subnet. Then, we present ML resource allocation approaches for instantiating a TN and a CN slice subnet.

Broadcasting supply distribution

RAN slice subnet instantiation is usually formulated as the difficulty where the resources of one or more BSs, i.e., spectrum, power, antennas, among others, must be shared between multiple slices [42]. In the literature, the RAN slicing difficulty has been tackled on two different levels: preparation and runtime. In the following, we discuss works dealing with RAN slicing at the planning level. At the runtime level, RAN slicing is realized through slice elasticity, which will be discussed in Section V.

At the planning level, RAN resources are allocated to each slice before its operation based on capacity and isolation requirements. In our scrutiny, we observed that works dealing with RAN slicing at the planning level fall into two categories: those applying a combined slice admission control and re- source allocation solution and those using slice traffics/resource demand prediction. Since ML solutions for the slice admission control difficulty have been introduced in Section III-B, in this section, we discuss relevant works that use ML for predicting traffics/resource usage for RAN slicing.

Gutter man et al. [43] proposed a metric for a slice named REVA, denned per Quos Class Identifier (QCI) and traffics direction. REVA measures the resource rate (in PRBs/sec) accessible for a Very Active bearer, i.e., a bearer that con- tenuously attempts to obtain more PRBs than a maximal fair share available. The authors then urbanized a prediction model for this metric and used it for slice provisioning. The work collected traces of RAN resource allocation from a custom-calculated new Long Term execution (LTE) tested under different multipart tradition patterns to build the model prediction. The authors then designed a muddied Long Short name recollection (LSTM) model to predict REVA tens of second in move on. The precision of the LSTM was Evaluated against the Autoregressive Integrated Moving Aver-age (ARIMA) model and traditional LSTM neural complex, showing that the proposed model outperforms ARIMA and LSTM by up to 31%. Finally, the authors designed a slice provisioning algorithm that exploits the prediction models to minimize costs for tune providers.

A complex slice admittance control joined with resource share guided by a forecasting module that predicts net- work slices traffics and user mobility patterns is presented by Sciancalepore et al. [44]. In their proposal, the authors assumed that traffics needs within a slice follow a periodic pattern, applying time-series forecasting based on the Holt- Winters technique to predict the aggregate traffics for every admitted slice. The authors also employed the Self-similar least-action human walk (SLAW) mobility model for user mobility prediction. Using traffics generated by this model, the authors developed a Markova chain to capture the mobility pattern of a user and assumed that a weighted combination of such patterns rejects the mobility of a tenant. The authors then employed an UL method to learn the weights of each tenant. Next, they shared the overall load predicted by the Holt-Winters method and the mobility model to derive the predicted amount of resources requested by the tenant under a BS. Finally, the authors designed a RL algorithm to perform admission control considering the SLA of the different tenants, their traffics usage, and user distribution. Performance evaluation was conducted using a MATLAB simulation with 7 BSs, 10 tenants, and 100 UEs per tenant distributed firmly. Results show that proper forecasting increases system utilization, especially as the number of complex slice requests and system capacity grows.

Sapavath et al. [45] studied the Sparse Bayesian Linear Regression (SBLR) and Support Vector Engine (SVM) tech- inquest to estimate and predict Channel State Information (CSI) to make a decision about broadcasting frequency slicing. The system model was composed of infrastructure providers that sublease their broadcasting frequency for Mobile Virtual Complex Operators (MVNOs) based on the requests coming from MVNOs and their SLAs. Depending on the demands and requirements, users are classier into three user groups (stationary, mobile, and indoor) and the infrastructure provider’s wireless resources are allocated to MVNOs to serve the users of individual groups. Given the end-user demands, RAN resource pool, the number of available antennas, and the total bandwidth of the broadcasting frequency slices, the solution assigns wireless resources for the slice considering the data rate of each user of the slice. This data rate, in turn, is computed based on the estimated CSI. The training dataset was acquired through pilot- based training and data augmentation. Performance evaluation focused mostly on the accuracy of the predictors and showed that SBLR results in better outcomes than SVM, demonstrating that this technique is less sensitive to sparse CSI information.

VNF Placement

The TN and/or CN Slice Subnet Instantiation difficulty is usually formulated as the placement of a set of VNFs towards the underlying physical infrastructure. This approach is a typical VNE difficulty reformulated to consider septic re- quirements of the 5G system such as Random Contact

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 10, No. 3, October : 2023

Memory (RAM), CPU, disk, bandwidth, and latency constraints, as well as node sharing. Indeed, in the VNF placement difficulty, given a physical complex G, representing the underlying physical infrastructure, and a virtual complex H, representing the slice, we have to embed the virtual onto the physical complex so that each virtual node m H is mapped onto a physical node in G and each virtual link m, n H is mapped to a loop- free physical path in G connecting the two physical nodes to which the virtual nodes m and n have been mapped [16]. The objective is to and an embedding with the least cost that stases all link and node capacity constraints. The cost may represent congestion, preference in terms of operator agreements, load balancing, or real cost of operation.

The most relevant works that use ML to solve the VNF placement difficulty formulate it as a Markov Decision Process (MDP) and solve it using DRL. Yan et al. [46] proposed a combined DRL with a neural complex structure based on graph convolution complex to solve the VNF placement difficulty. In their proposal, states are represented by eight attributes: the number of CPU resources over all nodes, the amount of bandwidth available in each node, the amount of free CPU currently available in each node, the amount of bandwidth not allocated in each node, a vector describing the embedding for the current slice request, the number of CPU and bandwidth resources needed by the current slice request, and the number of unallocated virtual nodes in the current request. To reduce the number of input features, links are not explicitly considered in the state representation. Instead, a Graph Convolution Complex (GCN), a Convolution Neural Complex (CNN) used to extract features from homogeneous graphs, is employed to automatically extract link features from the physical complex. The action taken by the RL agent is the index of the physical node in which to place a septic VNF of the slice. This way of modeling the actions breaks the process of placing one slice in a sequence of VNF placements and reduces the size of the action space to the number of physical nodes. The reward function combines the acceptance ratio, the placement cost, and the load balance. The solution was evaluated through simulation using a substrate complex topology generated following the Waxman random graph. CPU and bandwidth resources of the substrate complex were uniformly distributed between 50 and 100 units, while slice requests were generated by a Poisson process.

Khaki et al. [47] also employed DRL and CNNs to improve the quality of a VNF placement heuristic. However, different from [46], the authors in [47] used a Relational GCN, which operates over heterogeneous graphs. The authors only consider resource-related features (CPU and bandwidth) to represent the system state, while the action is represented by a binary variable used to keep the same placement of the current VNF or to modify it based on a computed heuristic. The objective of the solution is to maximize the infrastructure provider revenue. The evaluation was performed through sum- elation on a complex topology following the Waxman random graph. CPU and bandwidth requests are drawn uniformly, as well as the number of VNFs in each request.

A Deep Deterministic Policy Gradient (DDPG) approach is employed by Quant et al. [48]. Different from [46] and [47], in [48], the state representation includes resource-related tares (CPU and bandwidth) and latency-related properties. The action taken by the DRL agent is represented by two sets of weights: one indicating the placement priority of each VNF in the slice request on each physical node and the other indicating the placement priority of each virtual link on each physical link. The reward function of action is modeled as the acceptance ratio. To assess the performance of the proposed approach, the authors employed simulations using a real-world complex topology with 24 nodes and 37 links. Link capacities are randomly chosen, the requested VNF resources are uniformly distributed, and virtual links are arbitrarily requested with bandwidth in the range of 1 Mbps to 40 Mbps and latency of 1 ms to 100 Mrs.

Ensuring that a DRL agent converges to an optimal poll- icy in the VNF placement difficulty is a challenge since its performance depends on the exploration of a huge number of states and actions. To overcome this difficulty, Estevez et al. [49] introduced the concept of Heuristically Assisted DRL, which combines a DRL algorithm based on Advantage Actor-Critic (A2C) and a

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 10, No. 3, October : 2023
GCN with a Power of two Choices heuristic to control the DRL convergence. The RL elements of the solution (i.e., state, action, and reward) follow the same approach in [46]. The performance evaluation is carried out through simulation with three data center types (edge, core, and cloud) and one slice type (ebb). Slice requests involve eve VNFs, and arrival rates follow three complex load conditions (under load, normal load, and critical load).

Mei et al. [50] handled the VNF placement difficulty by creating a VNF pool. This pool integrates all individual VNFs distributed in the complex d o chiefs, providing a variety of complex abilities to meet the requirements of Vehicle-to- Everything (V2X) tunes. An Intelligent Control Layer is re-possible for orchestrating the available VNF (e.g., allocating VNFs and complex resources to complex slices). The solution intends to support the deployment of VNFs on remote and edge clouds by using Deep Q-Complex (DQN) with CNNs. The solution was evaluated through simulation with an urban scenario based on the Manhattan grid layout and two types of Vehicle-to-Vehicle (V2V) tunes: traffics safety and efficiency tune and autonomous driving-related tune.

Kabila et al. [51] tackled the multi-do chief slicing as a multi-substrate VNF difficulty. In their proposal, a DRL algorithm selects the optimal set of infrastructure providers among all the feasible candidates to maximize the revenue-to- cost ratio for deploying the slice requests. The DRL algorithm is based on a NN that takes as input an M x N feature matrix, where M is the number of infrastructure providers and N is the number of extracted features. The latter rejects the attributes of both the slice request and substrate complex. The NN was trained offline using demands of the size of 500 requests per epoch, with the request delay uniformly distributed between 1 to 200 units. The evaluation considered an online scenario where the request arrival follows a Poison distribution. A comparison with a combinatorial scheme showed that the DRL algorithm presents a better performance, especially in the presence of high request arrival rates.

Antacid et al. [52] proposed an NS strategy that uses FL to support slice allocation through VNF placement in distinct tune areas with different costs and processing and storage capabilities. In their proposal, UEs are mapped into three slice classes (high-rate communications; highly dynamic, low-rate, and delay-tolerant communications; and URLLC), and a FL framework is employed to foresee the UEs' demand of each tune class. The aim is to use the forecast UEs' demand to provide a VNF placement that maximizes the infrastructure provider revenue while improving the end user's Woe. The FL framework applies ML models trained at the UE level, and then a central layer aggregates to improve the global erudition model. To capture the UE request behavior, the authors use Prospect Theory (PT). The latter aims at evaluating a prospect (tune area) denned over a set of outcomes (UE tune completion time) and the probability associated with each of them. The proposed framework was evaluated through simulation involving eight different areas with processing and storage capacities, VNF types, and costs uniformly distributed. The VNFs requests were modeled by using the Movie Lens dataset.

Panayiotis et al. [53] focused on the TN Slice Subnet Instantiation difficulty. The objective is to dine a transport path considering a multi-do chief complex slice, which could span many paths. In this context, the authors work on the Quality of Transmission (Sot) estimation for sliceable optical complex. The authors examine centralized and distributed NN-based Sot estimation model for sliceable optical net- works. The objective is to and Sot model(s) that are new- tuned to the diverse requirements of each slice. The centralized difficulty is formulated as a multiclass classier trained with global complex information while the distributed difficulty is formulated as a set of binary classiness, each of them trained according to data that is relevant to a single type of slice. The results show that the distributed Sot model performs better than the centralized model, being independent of the number of slice types. Table III summarizes the chief characteristics of the literature related to ML applied to ML difficulty in the commissioning phase.

ML FOR NS IN THE OPERATION PHASE

The complex slice operation phase requires intense manage-men activity in run-time. In addition to

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 10, No. 3, October : 2023

activating the complex slice instance provisioned in the commissioning phase, the operation phase also cares about the supervision, performance reporting, medication, and resource capacity planning [29]. Therefore, the state-of-the-art brings several ML approaches for various complex slice operation tasks. In our review, we and out that ML is often adopted to solve the following NS difficulty in the operation phase: complex slice elasticity, user admission control; traffics classification and prediction; anomaly detection, task offloading, congestion control, RAT selection, and NS with mobility. This section details how relevant works in the state-of-the-art tackle each difficulty. Table IV summarizes the chief characteristics of the literature related to ML applied to LCM difficulty in the operationphase.

TABLE III
SUMMARY OF ML-APPROACHES FOR NS DIFFICULTY IN THE COMMISSIONING PHASE.

Ref.	NS Difficulty	Erudition Paradigm	Erudition Method	Resource Type	Complex Segment	Performance Evaluation
[43]	Broadcasting sharing with traffic prediction	SL	Modified LSTM	Complex	RAN	Experimental - Traces collected from a LTE testbed
[44]	Broadcasting sharing with traffic prediction	Time-series, UL, and RL	Holt-Winters, Customized RL algorithm	Complex	RAN	Simulation - SLAW mobility model
[45]	Broadcasting sharing with traffic prediction	SL	SBLR, SVM	Complex	RAN	Simulation - Pilot-based symbols
[46]	VNF placement	RL	GCN	Computing, complex	TN, CN	Simulation - Poisson distribution for slice requests
[47]	VNF placement	RL	Relational GCN	Computing, complex	TN, CN	Simulation - Uniform distribution for slice requests
[48]	VNF placement	RL	DDPG	Computing, complex	TN, CN	Simulation - Uniform distribution for slice requests
[49]	VNF placement	RL	A2C, GCN, heuristic	Computing, complex	Cloud, CN, Edge	Simulation - Customized arrival rate
[50]	VNF placement	RL	DQN, CNN	Computing	Cloud, Edge	Simulation - Customized V2V tunes
[51]	VNF placement	RL	CNN	Computing, complex	TN, CN	Simulation - Poisson distribution for slice requests
[52]	VNF placement	FL	PT	Computing, storage	CN	Simulation - VNF request generated from real-world dataset
[53]	Path configuration	SL	NN	Complex	TN	Simulation - Poisson distribution for connection request

Complex Slice Elasticity

Complex slice elasticity embraces run-time tasks to modify the current slice deployed to support a user demand or apply- action requirement. Li et al. [54] brought solid contributions to reviewing the background of DRL and its usage for resource running in NS. The work follows two chief scenarios: (i) resource running for RAN; and (ii) priority scheduling in typical VNF. Relying on DQL, the authors proposed an approach based on allocating resources regarding the users' activity. Such solution performed better than other intuitive approaches, such as demand prediction, no slicing, and hard slicing.

I et al. [55] presented an enhancement to the Applica- ability of DQL. The authors show how to allocate/reallocate limited spectrum across slices by improving the calculation and approximation of the Q-value function. The authors argue that their approach is suitable for NS tasks, having faster con- mergence and better performance than typical DQL. However, they point out that there is still space for research in aspects such as SLA assurance.

Li et al. [56] proposed an algorithm for end-to-end NS resource allocation based on DQN. However, we at this work into the complex slice operation phase due to its contra- bunion to slice elasticity, which assumes slice instantiation and execution. The authors presented a framework for 5G resource allocation, considering wireless resources on RAN and VNF on CN. A DQN algorithm uses the feedback from the environment dynamically and in real-time to update the wireless resources and map the tune links. Simulations support the results in terms of contact rate.

Boozed et al. [57] demonstrated an intelligent solution for dynamic capacity allocation in an end-to-end complex slice with multiple cloud-enabled virtualized segments for a video replay tune. A RL algorithm is used with predictive models (trend-based and parametric methods) for state estimation. Authors argue that the predictive models with RL can manage the elasticity through a servicing gateway and Web servers and cooperate to enhance the global system efficiency.

Guan et al. [58] proposed a hierarchical resource man- argument framework that utilizes DRL to perform resource adjustment within admitted end-to-end slices. The proposed framework introduces 1) multiple local resource manager to deal with the demand changes in resource requirements for an individual slice; and 2) a global resource manager to control the local resource managers. The local resource manager executes a DQL algorithm, where states represent the current tune quality satisfaction, actions denote whether slice adapt- tin is required, and reward is denied as the revenue obtained by adjusting resources minus the resource consumption cost and operational cost. Evaluation is performed using simulation on complex and computing resources.

Indeed, because of its edibility, dynamism, and high applicability for large-scale difficulty, ML techniques apply to the most diverse complex slice elasticity issues, such as complex performance and overall resource optimization and Quos guarantee. The vast majority of reviewed work in our scrutiny concentrates on this category (Table IV presents a summary of all of them). In addition, most of these difficulty sit on the RAN segment, and DRL is the most selected ML technique to deal with them [59]–[70], followed by supervised erudition [71]–[77]. The authors in [30] use a Deep Neural Complex (DNN) to decide on complex slice re configuration in a Metro-Core optical complex.

purchaser entry manage

The user admission control NS difficulty aggregates articles regarding challenges in deciding whether a new user, upon request, should be added to a running complex slice or not. The difference between user admission control and slice admission control is that, in the former, the request is for including a new user into an instantiated and running complex slice. Overall, user admission control is a process that ponders what is being requested vs. what is or will be available to be consumed (e.g., bandwidth, computing, storage, and broadcasting spectrum). Admitting new users into a running complex slice means that the operator commits to the availability of resources (e.g., spectrum and bandwidth) to serve all the users hosted in the slice.

In 5G complex, typically, user requirements may change over time (e.g., depending on the running applications) and a single UE may connect to up to 8 complex slices simulate- onerously [78]. Therefore, ML techniques act in this decision- making based on the time-varying user requirements, possible resource allocation for newcomers, eventual slice elasticity, and long-term SLA holding, for example.

Shame and Kudeshia [79] focused on the RAN and consider- erred three different generic slice templates: match, URLLC, and ebb. Based on a muddied version of classical DQL, users are allocated/reallocated to slices regarding their current needs. In the literature, this type of difficulty is also called Slice Selection. However, we consider it fraction of user admission since the process requires aggregating new users into running slices. In [79], the authors set up an experiment simulation scenario with multiple MVNOs sharing virtual BS resources. The experiment sets 30 MHz bandwidth for each virtual BS, distributing it among 100 users. Each one of the users has requirements sting them in at least one of the three generic slice templates (match, URLLC). Results show that the authors' proposal keeps a high average user satisfaction score during the experiment.

Nasser and Wilma [80] considered a 5G scenario and discussed the limitation of resources at the complex edge, specifically at fog nodes supporting vehicular and smart-city complex. The NS proposal includes creating a cluster of fog nodes with a controller, referred to as Edge Controller (EC), responsible for efficiently managing resources. The EC uses DRL to adapt to optimal slicing policies, performing admission control tasks (e.g., serving or avoiding new users, serving or avoiding septic requests) towards load balancing, saving resources, and denying tasks better performed in the cloud. The authors evaluate the proposal's suitability for the edge complex using simulations.

Traffic classification and guess

This category embraces works on the slice run-time using ML techniques for traffic classification and/or prediction. Traditionally, classifying complex traffic involves three com- mom approaches: port-based, Deep Packet Scrutiny (DPI), and statistical. ML techniques are especially appropriate for statistical approaches, which classify the traffic according to, for example, the packets' size and transmission direction.Therefore, the state-of-the-art presents NS methods based on classifying the traffic to infer running applications, predict bandwidth, and dynamically allocate/reallocate resources.

Le et al. [81] presented early-state contributions for the future Self-Organized Complex (SONs) NS. The authors aim to build an architecture for NS based on mobile broadband traffic classification. Based on past contributions working on big data, ML, and SDN/NFV, the authors use K-means as an UL algorithm for clustering mobile applications, resulting in three slices (0.5Mbps, 1Mbps, 3Mbps). They also apply several SL techniques (e.g., Naive Bays, SVM, NN) for classifying new coming traffic bows into the three distinct slices.

Results show high accuracy in traffic classification and therefore promising early-state contributions.

The authors in [82] used an FL method based on Key Performance Indicator (KPI) data collection (e.g., complex traffic) at virtualized Central Units (CUs) to chieftain dis- tribute local datasets, referred to as Mini-Datasets. These distributed Mini-Datasets compose the FL model for resource allocation with long-term SLA constraints. In this context, the authors have another complementary publication [83] focusing on the energy efficiency perspective of their approach.

Terra et al. [84] presented an analysis of explainable Artificial Intelligence (XAI) methods applied to telecomm- inaction complex. XAI methods are applied to analyze the cause of SLA violation prediction made in 5G complex. The proposal analyzes the explanation directly generated from the SLA violation prediction instead of expert knowledge. Local Interpretable Model-Agnostic Explanations (LIME), Shapely Additive Explanations (SHAP), Permutation Importance (PI), and Extreme Gradient Boosting (XGBoost) XAI methods are used to analyze SLA violation prediction causes, and these methods are further compared among them.

TABLE V
SUMMARY OF ACRONYMS.

Acronym	**Definition**	**Acronym**	**Definition**
3GPP	3rd Generation Fractionnership Project	N3AC	Neural Complex Admission Control
5G	fifth-generation	NFV	Complex Function Virtualization
6G	sixth-generation	NN	Neural Complex
A2C	Advantage Actor-Critic	NS	Complex Slicing
ACM	Association for Computing Enginery	NS-3	Complex Simulator 3
AI	Artificial Intelligence	NSS	Complex Slice Subnet
ARIMA	Autoregressive Integrated Moving Average	O-RAN	Open RAN
BE	Best Effort	ONETS	Online Complex Slice Broker
BS	Base Station	PI	Permutation Importance
CN	Core Complex	PRB	Physical Resource Block
CNN	Convolutional Neural Complex	PT	Prospect Theory
CPU	Central Processing Unit	QCI	QoS Class Identifier
C-RAN	Cloud-RAN	QoE	Quality of Experience
CSI	Channel State Information	QoS	Quality of Tune
CU	Central Unit	QoT	Quality of Transmission
DDPG	Deep Deterministic Policy Gradient	RAM	Random Contact Memory
DQL	Deep Q-Erudition	RAN	Broadcasting Contact Complex
DQN	Deep Q-Complex	RAT	Broadcasting Contact Technology
DL	Deep Erudition	RIC	RAN Intelligent Controller
DNN	Deep Neural Complex	RL	Reinforcement Erudition
DPI	Deep Packet Scrutiny	RU	Broadcasting Unit
DRL	Deep Reinforcement Erudition	SARSA	State-Action-Reward-State-Action
DU	Distributed Unit	SDN	Software-Defined Complex
EC	Edge Controller	SDO	Standards Developing Organization
FL	Federated Erudition	SHAP	SHapely Additive Explanations
F-RAN	Fog-Broadcasting Contact Complex	SL	Supervised Erudition
GCN	Graph Convolutional Complex	SLA	Tune Level Agreement
H2H	Human-to-Human	SLAW	Self-similar least-action human walk
HDBSCAN	Hierarchical Density Based Spatial Clustering	SMDP	Semi-Markov Decision Process
IEEE	Institute of Electrical and Electronics Engineers	SMO	Tune Running and Orchestration
IoT	Internet of IoT	SON	Self-Organized Complex
IP	Internet Protocol	SVM	Support Vector Engine
KPI	Key Performance Indicator	TL	Transfer Erudition
LCM	Being Series Running	TN	Transport Complex
NLP	Natural Language Processing	TRPO	Trust Region Policy Optimization
LIME	Local Interpretable Model-Agnostic Explanations	UCB	Upper Confidence Bound

LSTM	Long Short Term Memory	UE	User Equipment
LTE	Long Term Evolution	UL	Unsupervised Erudition
M2M	Engine-to-Engine	URLLC	ultra-Reliable and Low-Latency Communications
MAB	Multi-Armed Bandit	V2V	Vehicle-to-Vehicle
MDP	Markov Decision Process	V2X	Vehicle-to-Everything
ML	Engine Erudition	VNE	Virtual Complex Embedding
mMTC	Engine Type Communication	VNF	Virtualized Complex Function
MPLS	Multiprotocol Label Switching	XAI	eXplainable Artificial Intelligence
MVNO	Mobile Virtual Complex Operator	XGBoost	Extreme Gradient Boosting
ZSM	Zero touch complex & Tune Running		

Sahib et al. [85] proposed a micro-tune-based expert- mental prototype with a regression tree algorithm to vale- date the impact of forecasting capabilities on the RAN slice- in running. The experimental prototype, based on the Open Air Interface deployment, collects data while managing several Iota devices. This data then forms a time series used to train the regression tree. The objective is to forecast the number of PRBs to be used by each slice to dynamically provision the optimum slicing ratio out of the available pool of PRBs. Results show that the forecasting model can increase substantially the throughput of the complex at the cost of increased computing resources utilization.

additional investigation

This subsection groups together relevant difficulty for the complex slice operation phase. However, in our research, no substantial amount of articles discussed an ML approach to solve them. In this sense, we present at least one publication approaching each difficulty. Refer to Table IV for a complete list of publications regarding the operation phase.

1) Anomaly Detection: AI-assisted anomaly detection is a classical research held in computer complex [86], [87]. Analyzing the complex behavior (e.g., based on KPIs such as packet loss and downlink delay) is a running task during the complex slice operation phase. In [88], the authors implemented an AI-based module for assisting administrators in detecting anomalies among tunes in slices deployed on a virtualized infrastructure. The solution of the authors, aiming to classify complex traffic, has three phases: (i) pre-processing and feature selection, (ii) clustering, and (iii) anomaly de-section. Data is modeled as a time series composed of the following features: number of lost packets per tune and user, uplink and downlink delay, Reference Signal Received Power, transfer protocol, and UE received bytes. In the second phase, the time series is processed by a Hierarchical Density Based Spatial Clustering (HDBSCAN) clustering algorithm, which divides the dataset into three groups: normal, moderate, and anomalous behavior. Such clusters are used to label the samples in the time series. Finally, in the third phase, the labeled dataset is used to train a feed-forward DNN to perform a classification task. After, the DNN is used to predict anomaly and assign a cluster to new data in real-time. Preliminary results using the Complex Simulator 3 (NS-3) discrete-event simulator show a high accuracy score. However, increasing the number of clusters and the algorithm granularity decreases the prediction performance.

2) Task Offloading: The task offloading difficulty category addresses articles regarding decision-making on the most suitable do chief to run a task (e.g., UE, cloud, fog). In this sense, the authors of [89] discuss the adaptive mode selection in Fog-RAN (F-RAN), which refers to the communication mode serving each UE (e.g., Cloud-RAN (C-RAN), fog-broadcasting contact point, device-to-device).

3) Congestion Control: Selected works approaching the congestion control difficulty with ML

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 10, No. 3, October : 2023
fall into the scenario of connection establishment for RAN and traffic congestion control in general for 5G/6G wireless complex. The authors in [90] argue that Engine-to-Engine (M2M) complex traffic may surpass Human-to-Human (H2H) in the future. However, current approaches for dealing with M2M traffic rely on legacy congestion control schemes, which will no longer suit the demand in 5G and beyond scenarios. Therefore, the authors propose an improved congestion control scheme based on RL.

4) RAT selection: Cellular complex adopting multiple different RAT impose the well-known RAT selection chalk- lunge [97]. The article [91] presents IRIS, a shared spectrum contact architecture for indoor neutral-host small cells. IRIS adopts a RL algorithm based on DDPG to dynamically price the cost of a broadcasting spectrum block in an indoor shared setting according to the previous price, tenants (operators) demands, acquisition costs, and neutral-host revenue target.

5) NS with Mobility: This difficulty category considers works discussing scenarios with mobility in terms that the UE is not static. To the best of our knowledge, the chief concerns, up to now (the date of this research), in the context of NS with mobility are coverage area [93]; content caching [92]; and slice migration (e.g., UE moves out of the coverage and needs reallocation to another slice) [95], [96]. Added et al. [94] propose and evaluate two DRL-based algorithms for the intelligent selection of triggers supporting NS mobility actions. Authors argue their approach is new by consideringusers mobility, tune mobility, and resource mobility among slices for slice, tune, and resource allocation. The run-time mobility decision-making process is evaluated considering the A2C, a hybrid DRL method combining value-based and policy-based approaches and DQN.

unhook do study issue AND prospect strategy

This section identifies and discusses a non-exhaustive set of open issues on ML for intelligent NS. The identified challenges result from our analysis of the preparation, com- visioning, and operation phases of the NS process, presented in sections III, IV, and V. Moreover, we highlight the chief gaps in the literature between requirements and proposals for intelligent NS.

smart paraphrase for NS supplies

Translation of tune prowls into NS requirements is a complex task that requires low-level complex slice con- gyration parameters, such as virtual engine parameters, complex configurations, topology, and protocols [98]. With the evolution of complex toward beyond 5G, the complexity of this task tends to increase [99]. Consequently, an intent layer will be required to translate tune prowls into slice requirements [100].

Intent-driven complex was conceived to enable apply-captions to express desired operational aims using high-level descriptive specifications known as intents [101]. Addressing this aim, however, poses several challenges, among them, denting rich semantics to express the intent of verticals [102]. Although the integration with AI technologies, and Natural Language Processing (NLP) in fractionicular, can bridge this gap, those technologies are still at their early stage and require further research efforts before being integrated into the complex slice LCM [100]. This research gap can be evincedin our scrutiny, where only one work [25] uses ML to bridge this gap.

Datasets and research duplicate

High-quality datasets are essential to support the extensive dissemination of ML in various application do chiefs. Intel- gent NS, according to our research scrutiny, is yet another area where openly available high-quality datasets are a research issue, regardless of the slice being series phase, as can be seen in the column Performance Evolution of Tables II, III, and IV. A directly related aspect of dataset availability is experiment reproduction. In effect, the unavailability of datasets for most of the research work is an obstacle to allowing experiment reproduction and, to some extent, the explain ability of theproposed solutions and their dissemination. In our scrutiny, most works (e.g., in [36], [37], [38], [40],[46] [47], [48], [49], [51], [53], [54], [55], [56], [58],

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 10, No. 3, October : 2023

among others) use data generated from simulation to evaluate their ML solutions. However, to evaluate the effectiveness of ML approaches when dealing with NS difficulty in practice, ex- tensile evaluations are needed taking more realistic scenarios into consideration. To this end, some works [26] [43] [72] [77]

TABLE IV
SUMMARY OF ML-APPROACHES FOR NS DIFFICULTY IN THE OPERATION PHASE.

Ref.	NS Difficulty	Erudition Paradigm	Erudition Method	Resource Type	Complex Segments	Performance Evaluation
[54]	Slice Elasticity	RL	DQL	Complex	RAN, CN	Simulation
[55]	Slice Elasticity	RL	DQL	Complex	RAN	Simulation
[56]	Slice Elasticity	RL	DQN	Complex	RAN, TN, CN	Simulation
[57]	Slice Elasticity	RL	QL	Complex	RAN, TN, CN	Not clear
[58]	Slice Elasticity	RL	DQL	Complex, computing	RAN, TN, CN	Simulation
[59]	Slice Elasticity	RL	DQN	Complex	RAN	Simulation
[60]	Slice Elasticity	RL	DRL	Complex	RAN	Simulation
[61]	Slice Elasticity	RL	DQL	Complex	RAN	Simulation
[62]	Slice Elasticity	RL	DDPG	Complex	RAN	Simulation
[63]	Slice Elasticity	RL	Duel. DNN, QL	Complex, computing, storage	C-RAN	Simulation
[64]	Slice Elasticity	RL	DRL	Complex	RAN	Simulation
[65]	Slice Elasticity	RL	DQN	Complex	RAN	Simulation
[66]	Slice Elasticity	RL	DQN	Complex	RAN	Simulation
[67]	Slice Elasticity	RL	DQN	Complex	RAN	Not clear
[68]	Slice Elasticity	RL	LSTM, A2C DRL	Complex	RAN	Simulation
[69]	Slice Elasticity	RL	DQL	Complex, computing	RAN, edge	Simulation
[70]	Slice Elasticity	RL	A2C	Complex	RAN	Simulation
[71]	Slice Elasticity	SL	DNN	Complex	RAN, CN	Real-world dataset
[72]	Slice Elasticity	SL	LSTM	Complex	RAN, CN	Exp. & real-world dataset
[73]	Slice Elasticity	SL	LSTM	Complex	RAN, TN	Real-world dataset
[74]	Slice Elasticity	SL	DNN, LSTM	Complex	RAN	Simulation
[75]	Slice Elasticity	SL	DNN	Complex, computing	RAN	Not clear
[76]	Slice Elasticity	SL	DNN	Complex, computing	RAN, TN	Real-world dataset
[77]	Slice Elasticity	SL	LSTM	Complex	RAN, TN	Experimental
[30]	Slice Elasticity	RL	DNN	Complex	TN	Simulation
[79]	User Adm. Control	RL	DQL	Complex	RAN	Simulation
[80]	User Adm. Control	RL	DQN	Complex, computing	RAN, edge	Simulation

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 10, No. 3, October : 2023

[81]	Traffic Prediction	UL, SL	Naive Bayes, SVM, NN	Complex	RAN, CN	Experimental
[82]	Traffic Prediction	FL	Non-zero sum	Complex	RAN	Real-world dataset
[84]	Traffic Prediction	XAI	XGBoost, SHAP	Complex	TN	Experimental
[85]	Traffic Prediction	SL	Regression Tree	Complex	RAN	Experimental
[88]	Anomaly Detection	UL, SL	HDBSCAN, DNN	Complex	RAN	Simulation
[89]	Task Offloading	RL	QL	Complex, computing	RAN	Simulation
[90]	Congestion Control	RL	TRPO	Complex	RAN	Simulation
[91]	RAT Selection	RL	DDPG	Complex	RAN	Exp. & Simulation
[92]	NS Mobility	RL	DQN	Complex, computing	F-RAN	Simulation
[93]	NS Mobility	SL	DNN	Complex	RAN, edge	Simulation
[94]	NS Mobility	RL	A2C, DQN	Complex, computing	RAN, edge	Simulation
[95]	NS Mobility	RL	A2C, LSTM	Complex	RAN	Simulation
[96]	NS Mobility	RL	QL	Complex	RAN	Simulation

[81] [84] [85] [91] create septic experimental test beds for validating their model or algorithm. Although such initiatives are important, data collected from test beds still misses the representative of the complexity and dynamicity of real-world mobile complex [103]. In addition, none of such works have made the collected data available for the research community, hindering and compromising the reproduction of the mental fractions deployed for validation purposes. Finally, 10% of the scrutiny end works [52] [71] [72] [73] [82] [76] use real- world complex data. Although such data are much richer and more representative than those generated from simulation or test beds, they still may suffer from noise, sparsely, and lack of label, which limits the ML algorithms that can be applied. In summary, rich and adequate data is still an issue for applying ML in NS difficulty.Suitability of the ML Technique for the Complex Slice Being Series Phase

While ML is an unquestionable enable for the realization of NS, it is impossible to and a single technique that completely addresses the requirements of all the complex slice LCM difficulty. Thus, an open research issue in ML-enhanced NS is the suitability of the ML techniques for the target complex slice being series phase with regard to, for example, granularity or timing [23]. In the preparation phase, as the slice does not exist, ML techniques using offline erudition can be applied to solve the difficulty of such phase. Indeed, the authors in [38] conclude that offline training solutions for the slice admission control difficulty require a training period before use but give the best results.

In the slice commissioning and operation phases, SL, which usually relies on offline erudition, has been usually applied to solve traffic prediction, traffic classification, and anomaly detection difficulty [43] [45] [81] [85] [88]. However,
elms that involve resource allocation, such as broadcasting resource sharing, VNF placement, and complex slice elasticity, have to make decisions on low scale and cannot afford for a period of training time [104]. Thus, some works [44] [57]
[89] [90] use classical RL algorithms with online training for these difficulty. However, resource running in NS usually involves multidimensional parameters, leading to a large state space and a low convergence rate to the optimal policy [104]. In practice, this means that, until the RL algorithm converges, it can make bad resource running decisions. Although DRL algorithms have been used to face this limitation (e.g., in [46], [47], [48], [49], [50], [51], [80], [54], [55], [56], [58], among

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 10, No. 3, October : 2023
others), DRL solutions present some shortcomings. First, DRL algorithms usually rely on DQN to encode state. However, an important component of DQN is a target complex, i.e., a copy of the estimated value function that is kept axed for some number of steps to stabilize erudition [105], [106]. This copy, in turn, prevents the algorithm from reacting fast to environment changes, a desired property of RL. More recently, other NN algorithms (e.g., in [68]) have been investigated to deal with this difficulty. Nevertheless, further investigations are required to determine their efficacy and generalization in the context of DRL. With this regard, TL has also been considered a possible solution [104]. Another difficulty with DRL algorithms is that NNs with multiple layers cannot explain the essential features that impudence their decisions or the impact of data bias on the uncertainty of outputs [107]. As complex slices are expected to host an increasing number of mission-critical tunes in beyond 5G, trust will become critical. Despite this need, our scrutiny identified only one work [84] addressing explainable ML-enhanced NS. Finally, it is important to highlight that DRL algorithms have a high demand for computing, memory, and energy resources [103]. Considering that beyond 5G complex will make pervasive use of intelligence [21], DRL algorithms that make more deficient use of computing and power resources are still an open issue.

End-To-End NS

NS is applied in challenging systems such as 5G and beyond 5G, Industry 4.0/5.0, and intelligent transportation systems. End-to-end NS is an essential requirement and current trend for these systems. However, in most scrutiny end works, ML support is focused on complex segment solutions (e.g., RAN [89], RAN + edge [80], TN [84], and RAN + CN [71], leaving end-to-end NS as an open research issue. We consider that an article effectively approaches end-to-end NS if it deals with the three complex segments (RAN, TN, and CN) completely. However, this issue is not a consensus in the literature. For example, in [77], the authors assume that the end-to-end can start inside the RAN, crosses a TN, and noshes at the frontier of a CN. The authors do not consider the front haul, i.e., fraction of the RAN is not sliced, nor the CN. In our scrutiny, only a few works effectively tackle the end-to-end slicing difficulty in the three segments (RAN, TN, and CN) [56] [57] [58], and the arts two only deal with complex resources. While [58] is a more comprehensive work considering complex and computing resources, the recital evaluation in this article is based on a small and implied simulation. The authors are focused on calling attention to the importance of ML-enabled NS in 6G and the challenges in the real-world implementation.

Slicing by segment with ML support is undoubtedly reel- event. Nevertheless, the end-to-end design must consider the interdependence of resource allocation and orchestration among complex segments. End-to-end intelligent slicing brings another level of complexity, which involves issues such as the need for a high-deficient (re)erudition process and coordination among multiple entities [58]. In this context, FL and other distributed erudition approaches may be relevant since the works can explore the spread processing capacity offered by edge computing and reduce the amount of information exchange.

unlock RAN intelligent portion

RANs are a fundamental fraction of the slicing process in 5G complex and Open RAN (O-RAN) is one the most relevant evolution aspect towards 6G in this segment. Not surprisingly, O-RAN architecture has AI and ML work bows in its design [108]. The O-RAN approach brings a new level of edibility for complex operators allowing them to deploy the RAN segment, potentially focusing on the business. NS has being considered a very important capability in the O-RAN context and has been already investigated in some articles [109]–[111]. O-RAN slicing with ML allows deficient RAN deployment to accomplish challenging user requirements regarding SLA, Foe, and user mobility.

The design, implementation, deployment, and evaluation of O-RAN with ML is a hot research topic and open research issue. In the context of O-RAN, an ML-based solution must be designed and implemented as a app. and/or a Rapp, depending on their time demands. While Capps run over a near-real-time RAN Intelligent Controller (RIC) (10ms to 1s), rap's run over a non-real-time RIC (more than 1s) [108]. Deployment and evaluation of caps and/or raps still depend on simulation (e.g., [109]), previously collected (and so non-interactive) datasets (e.g., [110]), or limited-size test beds using early-stage RICs (e.g., [111]). In fact, even the optimized deployment and operation of the RICs components are challenging since they are a new software platform still under development.

as of proposition to survive out

Based on the text presented in the previous sections, it is clear that several theoretical works are using AI and ML, considering the being series phases of NS. However, sciatic research with practical and experimental approaches to NS isstill in the beginning. As mention previously, most works use only simulation for validating their proposals, while some works focus on real-world traces or datasets, which is very useful for ML-based approaches. However, they also face hard issues such as information from outdated pre-NS technologies (LTE/4G, for instance) or data with low statistical bearing. Asexpected, only few works [26] [43] [72] [77] [81] [84] [85] [91] have already accepted the challenge of evaluate an
ML-based approach in an experimental tested. In this con- text, not only sciatic but also technological issues become relevant. For example, technological advances in telecomm- negations are increasingly based on native cloud computing platforms. Nevertheless, these platforms were not designed to support telecommunication tunes natively. Indeed, in our scrutiny, only the work [85] has validated its proposal using a micro-tune-based RAN experimental prototype.

Pushing the frontier of science by integrating theoretical advances in AI and ML with practical solutions for NS is an open issue that needs further investigation and advance efforts.

FINAL CONSIDERATIONS

This scrutiny focused on presenting NS with ML research contributions. The contributions are organized by the phases of the slice being series as denied by standardization organizations (preparation, commissioning, and operation phases), aiming to identify trends and correlated contributions for the different slicing phases. The contributions are rigorous in the 5G do chief, with few NS solutions applied to other areas. Specifically, in the 5G do chief, the end-to-end solution is a trend not yet fully explored, and ML is being extensively used to provide intelligence for segmented solutions. 5G end- to-end NS approach allows a global view of the resource allocation difficulty allowing for optimizing resource sharing aiming, for instance, to improve operation, achieve deficient running, and optimize operational expenditure. Although 5G end-to-end slicing is essential for tune providers and telecommunications operators, the scrutiny end articles primarily focus on slicing and optimizing segments like RAN and the CN.

We have observed that ML is already being investigated to solve several tasks in slice preparation, commissioning, and operation. In this context, different ML techniques and algorithms have been employed, chiefly the ones popularized in the last decades, such as CNN, GCN, LSTM, DRL, and XGBoost. ML has exhibited satisfactory or promising results in many automation tasks in the slice being series, which is critical to provide many berets related to the concept of Zero touch complex & Tune Running (ZSM). However, the practical and wide adoption of ML-enabled NS still faces several challenges. Some of these challenges, such as large and open datasets and the explain ability of ML-based solutions, are already being tackled by the academy and industry, which can count on the experience from other areas such as computer vision and natural language processing. However, other issues, such as the demand for short-time for model training, energy-efficient ML solutions, and distributed computation of ML models, still need much investigation. AI and ML are also evolving intensely, giving rise to new models, algorithms, techniques, and even hardware architectures. Traditionally, these novelties are not designed or tested arts in complex.

However, they must be imported and sometimes adapted in NS, for example.

Finally, we highlight the availability of various multi- technology (SDN, wireless, It, slicing, and others) test bedsworldwide for untried research expansion. These test beds, in most cases, innately facilitate experiment repro- diction using openly available software to control the expel- impend and having the ability to create experimental datasets. An essential point for researchers would be to evaluate to what extent these test beds can be used for developing and validate research results in bright NS.

REFERENCES

ITU-T, "New Tunes and Capabilities for Complex 2030: Description, Technical Gap and Performance Target Analysis," Available online: https://www.itu.int/dms_pub/itu-t/opb/fg/T-FG-NET2030-2019-SUB. G2-PDF-E.pdf, 2019, contacted on 20 June 2022.

[1] GSMA, "E2E Complex Slicing Architecture," Available online: https: //www.gsma.com/newsroom/wp-content/uploads/NG.127-v1.0-2.pdf, 2021, contacted on 20 June 2022.

ITU, "IMT Vision - Framework and overall objectives of the future development of IMT for 2020 and beyond," Available online: https://www.itu.int/dms_pubrec/itu-r/rec/m/R-REC-M. 2083-0-201509-I!!PDF-E.pdf, 2016, contacted on 25 March 2022.

NGMN Alliance, "Description of Complex Slicing Concept," Available online: https://ngmn.org/wp-content/uploads/160113_ NGMN_Complex_Slicing_v1_0.pdf, 2016, contacted on 27 December 2021.

Ordonez-Lucena et al., "Complex Slicing for 5G with SDN/NFV: Concepts, Architectures, and Challenges," IEEE Communications Mag- azine, vol. 55, no. 5, pp. 80–87, 2017.

X. Foukas et al., "Complex Slicing in 5G: Scrutiny and Challenges," IEEE Communications Magazine, vol. 55, no. 5, pp. 94–100, 2017.

X. Shen et al., "AI-Assisted Complex-Slicing Based Next-Generation Wireless Complex," IEEE Open Journal of Vehicular Technology, vol. 1, pp. 45–66, 2020.

I. Afolabi et al., "Complex Slicing and Softwarization: A Scrutiny on Principles, Enabling Technologies, and Solutions," IEEE Communica- tions Scrutiny & Tutorials, vol. 20, no. 3, pp. 2429–2453, 2018.

A. Kaloxylos, "A Scrutiny and an Analysis of Complex Slicing in 5G Complex," IEEE Communications Standards Magazine, vol. 2, no. 1, pp. 60–65, 2018.

S. Zhang, "An Overview of Complex Slicing for 5G," IEEE Wireless Communications, vol. 26, no. 3, pp. 111–117, 2019.

L. U. Khan et al., "Complex Slicing: Recent Advances, Taxonomy, Requirements, and Open Research Challenges," IEEE Contact, vol. 8, pp. 36 009–36 028, 2020.

A. A. Barakabitze et al., "5G complex slicing using SDN and NFV: A scrutiny of taxonomy, architectures and future challenges," Computer Complex, vol. 167, p. 106984, 2020.

M. Chahbar et al., "A Comprehensive Scrutiny on the E2E 5G Complex Slicing Model," IEEE Transactions on Complex and Tune Manage- ment, vol. 18, no. 1, pp. 49–62, 2021.

J. Ordonez-Lucena et al., "On the Rollout of Complex Slicing in Carrier Complex: A Technology Radar," Sensors, vol. 21, no. 23, 2021.

S. Wijethilaka and M. Liyanage, "Scrutiny on Complex Slicing for Internet of Things Realization in 5G Complex," IEEE Communications Scrutiny & Tutorials, vol. 23, no. 2, pp. 957–994, 2021.

S. Vassilaras et al., "The Algorithmic Aspects of Complex Slicing," IEEE Communications Magazine, vol. 55, no. 8, pp. 112–119, 2017.

R. Su et al., "Resource Allocation for Complex Slicing in 5G Telecom- munication Complex: A Scrutiny of Principles and Models," IEEE Complex, vol. 33, no. 6, pp. 172–179, 2019.

F. Debbabi et al., "Algorithmics and Modeling Aspects of Complex Slicing in 5G and Beyonds Complex: Scrutiny," IEEE Contact, vol. 8, pp. 162 748–162 762, 2020.

C. Ssengonzi, O. P. Kogeda, and T. O. Olwal, "A scrutiny of deep reinforcement erudition application in 5G and beyond complex slicing and virtualization," Array, vol. 14, p. 100142, 2022.
V. P. Kafle et al., "Consideration On Automation of 5G Complex Slicing with Engine Erudition," in 2018 ITU Kaleidoscope: Engine Erudition for a 5G Future (ITU K), 2018, pp. 1–8.
W. Wu et al., "AI-Native Complex Slicing for 6G Complex," IEEE Wireless Communications, vol. 29, no. 1, pp. 96–103, 2022.
D. Bega et al., "AI-Based Autonomous Control, Running, and Orchestration in 5G: From Standards to Algorithms," IEEE Complex, vol. 34, no. 6, pp. 14–20, 2020.
R. Li et al., "Intelligent 5G: When Cellular Complex Meet Artificial Intelligence," IEEE Wireless Communications, vol. 24, no. 5, pp. 175– 183, 2017.
R. Boutaba et al., "A Comprehensive Scrutiny on Engine Erudition for Complex: Evolution, Applications and Research Opportunities," Journal of Internet Tunes and Applications, vol. 9, no. 1, 2018.
H. Yang et al., "Automatic guarantee scheme for intent-driven complex slicing and reconfiguration," Journal of Complex and Computer Appli- cations, vol. 190, p. 103163, 2021.
V. Sciancalepore et al., "ONETS: Online Complex Slice Broker From Theory to Practice," IEEE Transactions on Wireless Communications,
vol. 21, no. 1, pp. 121–134, 2022.
G. Kibalya et al., "A reinforcement erudition based approach for 5G complex slicing across multiple dochiefs," in 2019 15th International Conference on Complex and Tune Running (CNSM). IEEE, 2019, pp. 1–5.
D. Bega et al., "Deepcog: Optimizing resource provisioning in complex slicing with ai-based capacity forecasting," IEEE Journal on Selected Areas in Communications, vol. 38, no. 2, pp. 361–376, 2020.
3GPP, "Running and orchestration; Concepts, use cases and re- quirements (Release 17)," Available online: https://www.3gpp.org/ftp/ Specs/archive/28_series/28.530/, 2021, contacted on 20 June 2022.
S. Troia, R. Alvizu, and G. Maier, "Reinforcement Erudition for Tune Function Chain Reconfiguration in NFV-SDN Metro-Core Optical Complex," IEEE Contact, vol. 7, pp. 167 944–167 957, 2019.
C. Jiang et al., "Engine Erudition Paradigms for Next-Generation Wireless Complex," IEEE Wireless Communications, vol. 24, no. 2, pp. 98–105, 2017.
B. Mao et al., "AI Models for Green Communications Towards 6G,"
IEEE Communications Scrutiny & Tutorials, vol. 24, no. 1, pp. 210–
247, 2022.
E. Zeydan and Y. Turk, "Recent Advances in Intent-Based Complex: A Scrutiny," in 2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring), 2020, pp. 1–5.
L. Velasco et al., "End-to-End Intent-Based Complex," IEEE Com- munications Magazine, vol. 59, no. 10, pp. 106–112, 2021.
N. Gritli, F. Khendek, and M. Toeroe, "Complex Slice Provisioning Taking Into Account Tenant Intents and Operator Policies," in 2021 IEEE Conference on Complex Function Virtualization and Software Defined Complex (NFV-SDN), 2021, pp. 167–173.
D. Bega et al., "Optimising 5G infrastructure markets: The business of complex slicing," in IEEE INFOCOM 2017 - IEEE Conference on Computer Communications, 2017, pp. 1–9.
D. Bega, M. Gramaglia, A. Banchs, V. Sciancalepore, and X. Costa- Pérez, "A Engine Erudition Approach to 5G Infrastructure Market Optimization," IEEE Transactions on Mobile Computing, vol. 19, no. 3, pp. 498–512, 2020.
S. Bakri, B. Brik, and A. Ksentini, "On using reinforcement erudition for complex slice admission control in 5G: Offline vs. online," International Journal of Communication Systems, vol. 34, no. 7, p. e4757, 2021.

G. Dandachi et al., "An Artificial Intelligence Framework for Slice Deployment and Orchestration in 5G Complex," IEEE Transactions on Cognitive Communications and Complex, vol. 6, no. 2, pp. 858– 871, 2020.
M. R. Raza et al., "Reinforcement Erudition for Slicing in a 5G Flexible RAN," Journal of Lightwave Technology, vol. 37, no. 20, pp. 5161– 5169, 2019.
B. Bakhshi, J. Mangues-Bafalluy, and J. Baranda, "R-Erudition-Based Admission Control for Tune Federation in Multi-dochief 5G Net- works," in 2021 IEEE Global Communications Conference (GLOBE- COM), 2021, pp. 1–6.
P. L. Vo et al., "Slicing the Edge: Resource Allocation for RAN Complex Slicing," IEEE Wireless Communications Letters, vol. 7, no. 6, pp. 970–973, 2018.
C. Gutterman et al., "RAN Resource Usage Prediction for a 5G Slice Broker," in Proceedings of the Twentieth ACM International Symposium on Mobile Ad Hoc Complex and Computing, 2019, p. 231240.
V. Sciancalepore et al., "RL-NSB: Reinforcement Erudition-Based 5G Complex Slice Broker," vol. 27, no. 4, pp. 1543–1557, 2019.
N. N. Sapavath, D. B. Rawat, and M. Song, "Engine Erudition for RF Slicing Using CSI Prediction in Software Defined Large-Scale MIMO Wireless Complex," IEEE Transactions on Complex Science and Engineering, vol. 7, no. 4, pp. 2137–2144, 2020, conference Name: IEEE Transactions on Complex Science and Engineering.
Z. Yan et al., "Automatic Virtual Complex Embedding: A Deep Rein- forcement Erudition Approach With Graph Convolutional Complex," IEEE Journal on Selected Areas in Communications, vol. 38, no. 6, pp. 1040–1057, 2020.
A. Rkhami, Y. Hadjadj-Aoul, and A. Outtagarts, "Learn to improve: A novel deep reinforcement erudition approach for beyond 5G complex slicing," in 2021 IEEE 18th Annual Consumer Communications Net- working Conference (CCNC), 2021, pp. 1–6.
P. T. A. Quang, Y. Hadjadj-Aoul, and A. Outtagarts, "A Deep Reinforcement Erudition Approach for VNF Forwarding Graph Em- bedding," IEEE Transactions on Complex and Tune Running, vol. 16, no. 4, pp. 1318–1331, 2019.
J. J. A. Esteves et al., "A heuristically assisted deep reinforcement erudition approach for complex slice placement," IEEE Transactions on Complex and Tune Running, pp. 1–1, 2021.
J. Mei, X. Wang, and K. Zheng, "Intelligent Complex Slicing for V2X Tunes Toward 5G," IEEE Complex, vol. 33, no. 6, pp. 196–204, 2019.
G. Kibalya et al., "A Reinforcement Erudition Based Approach for 5G Complex Slicing Across Multiple Dochiefs," in 2019 15th International Conference on Complex and Tune Running (CNSM), 2019, pp. 1–5.
R. Fantacci and B. Picano, "When Complex Slicing Meets Prospect Theory: A Tune Provider Revenue Maximization Framework," IEEE Transactions on Vehicular Technology, vol. 69, no. 3, pp. 3179–3189, 2020.
T. Panayiotou et al., "Centralized and Distributed Engine Erudition- Based QoT Estimation for Sliceable Optical Complex," in 2019 IEEE Global Communications Conference (GLOBECOM), 2019, pp. 1–7.
R. Li et al., "Deep Reinforcement Erudition for Resource Running in Complex Slicing," IEEE Contact, vol. 6, pp. 74 429–74 441, 2018.
C. Qi et al., "Deep Reinforcement Erudition With Discrete Normalized Advantage Functions for Resource Running in Complex Slicing," IEEE Communications Letters, vol. 23, no. 8, pp. 1337–1341, 2019.T. Li, X. Zhu, and X. Liu, "An End-to-End Complex Slicing Algorithm Based on Deep Q-Erudition for 5G Complex," IEEE Contact, vol. 8, pp. 122 229–122 240, 2020.
M. Bouzid et al., "Cooperative AI-based E2E Complex Slice Scaling," in IEEE INFOCOM 2019 - IEEE Conference on Computer Communi- cations Workshops (INFOCOM WKSHPS), Apr. 2019, pp. 959–960.

Guan, Wanqing and Zhang, Haijun and Leung, Victor C. M., “Cus- tomized Slicing for 6G: Enforcing Artificial Intelligence on Resource Running,” IEEE Complex, vol. 35, no. 5, pp. 264–271, 2021.
G. Sun et al., “Autonomous resource provisioning and resource cus- tomization for mixed traffics in virtualized broadcasting contact complex,” IEEE Systems Journal, vol. 13, no. 3, pp. 2454–2465, 2019.
A. Aslan, G. Bal, and C. Toker, “Dynamic resource running in next generation complex with dense user traffic,” in 2020 IEEE In- ternational Black Sea Conference on Communications and Complex (BlackSeaCom), 2020, pp. 1–6.
C. Zhang, M. Dong, and K. Ota, “Vehicular Multi-slice Optimization in 5G: Dynamic Preference Policy using Reinforcement Erudition,” in GLOBECOM 2020 - 2020 IEEE Global Communications Conference, 2020, pp. 1–6.
K. Yu et al., “A Reinforcement Erudition Aided Decoupled RAN Slicing Framework for Cellular V2X,” in GLOBECOM 2020 - 2020 IEEE Global Communications Conference, 2020, pp. 1–6.
W. Guan, H. Zhang, and V. C. Leung, “Slice Reconfiguration based on Demand Prediction with Dueling Deep Reinforcement Erudition,” in GLOBECOM 2020 - 2020 IEEE Global Communications Conference, 2020, pp. 1–6.
Y. Abiko et al., “Broadcasting Resource Allocation Method for Complex Slicing using Deep Reinforcement Erudition,” in 2020 International Conference on Information Complex (ICOIN), 2020, pp. 420–425.
M. Zambianco and G. Verticale, “Spectrum allocation for complex slices with inter-numerology interference using deep reinforcement erudition,” in 2020 IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Broadcasting Communications, 2020, pp. 1–7.
G. Sun et al., “Autonomous Resource Slicing for Virtualized Vehicular Complex With D2D Communications Based on Deep Reinforcement Erudition,” IEEE Systems Journal, vol. 14, no. 4, pp. 4694–4705, 2020.
I. Vilà et al., “A multi-agent reinforcement erudition approach for capacity sharing in multi-tenant scenarios,” IEEE Transactions on Vehicular Technology, vol. 70, no. 9, pp. 9450–9465, 2021.
J. Mei, X. Wang, and K. Zheng, “Semi-Decentralized Complex Slicing for Reliable V2V Tune Provisioning: A Model-Free Deep Re- inforcement Erudition Approach,” IEEE Transactions on Intelligent Transportation Systems, pp. 1–13, 2021.
G. Yang et al., “Two-tier resource allocation in dynamic complex slicing paradigm with deep reinforcement erudition,” in 2019 IEEE Global Communications Conference (GLOBECOM), 2019, pp. 1–6.
Y. Xu et al., “Constrained Reinforcement Erudition for Resource Allocation in Complex Slicing,” IEEE Communications Letters, vol. 25, no. 5, pp. 1554–1558, 2021.
D. Bega et al., “AZTEC: Anticipatory Capacity Allocation for Zero- Touch Complex Slicing,” in IEEE INFOCOM 2020 - IEEE Conference on Computer Communications, 2020, pp. 794–803.
K. Abbas et al., “Complex Slice Beingseries Running for 5G Mobile Complex: An Intent-Based Complex Approach,” IEEE Contact, vol. 9, pp. 80 128–80 146, 2021.
C. Zhang, M. Fiore, and P. Patras, “Multi-tune mobile traffic forecasting via convolutional long short-term memories,” in 2019 IEEE International Symposium on Measurements and Complex (M and N), 2019, pp. 1–6.
S. Matoussi et al., “Deep Erudition based User Slice Allocation in 5G Broadcasting Contact Complex,” in 2020 IEEE 45th Conference on Local Computer Complex (LCN), 2020, pp. 286–296.
X. Li et al., “5Growth: An End-to-End Tune Platform for Automated Deployment and Running of Vertical Tunes over 5G Complex,” IEEE Communications Magazine, vol. 59, no. 3, pp. 84–90, 2021.
H. Chergui and C. Verikoukis, “OPEX-Limited 5G RAN Slicing: an Over-Dataset Constrained Deep Erudition Approach,” in ICC 2020 - 2020 IEEE International Conference on Communications (ICC),

2020, pp. 1–6.
E. H. Bouzidi et al., “Online based erudition for predictive end-to- end complex slicing in 5G complex,” in ICC 2020 - 2020 IEEE International Conference on Communications (ICC), 2020, pp. 1–7.
3GPP, “Running and orchestration; Concepts, use cases and re- quirements (Release 16),” Available online: https://www.3gpp.org/ftp/ . Shome and A. Kudeshia, “Deep Q-erudition for 5G complex slicing with diverse resource stipulations and dynamic data traffic,” in 2021 International Conference on Artificial Intelligence in Information and Communication (ICAIIC), Apr. 2021, pp. 134–139.
A. Nassar and Y. Yilmaz, “Deep Reinforcement Erudition for Adaptive Complex Slicing in 5G for Intelligent Vehicular Systems and Smart Cities,” IEEE Internet of Things Journal, vol. 9, no. 1, pp. 222–235, 2022.
L.-V. Le et al., “SDN/NFV, Engine Erudition, and Big Data Driven Complex Slicing for 5G,” in 2018 IEEE 5G World Forum (5GWF), 2018, pp. 20–25.
H. Chergui, L. Blanco, and C. Verikoukis, “CDF-Aware Federated Erudition for Low SLA Violations in Beyond 5G Complex Slicing,” in ICC 2021 - IEEE International Conference on Communications, 2021, pp. 1–6.
H. Chergui et al., “Zero-Touch AI-Driven Distributed Running for Energy-Efficient 6G Massive Complex Slicing,” IEEE Complex, vol. 35, no. 6, pp. 43–49, 2021.
A. Terra et al., “Explainability Methods for Identifying Root-Cause of SLA Violation Prediction in 5G Complex,” in GLOBECOM 2020 - 2020 IEEE Global Communications Conference, Dec. 2020, pp. 1–7.
N. Salhab et al., “Autonomous complex slicing prototype using engine-erudition-based forecasting for broadcasting resources,” IEEE Com- munications Magazine, vol. 59, no. 6, pp. 73–79, 2021.
Y. Sani et al., “An overview of neural complex use in anomaly intrusion detection systems,” in 2009 IEEE Student Conference on Research and Development (SCOReD), 2009, pp. 89–92.
K. Abdelli et al., “Engine-erudition-based anomaly detection in optical fiber monitoring,” Journal of Optical Communications and Complex, vol. 14, no. 5, pp. 365–375, 2022.
V. A. Cunha et al., “5Growth: Secure and Reliable Complex Slicing for Verticals,” in 2021 Joint European Conference on Complex and Communications 6G Summit (EuCNC/6G Summit), 2021, pp. 347–352.
H. Xiang et al., “Mode selection and resource allocation in sliced fog broadcasting contact complex: A reinforcement erudition approach,” IEEE Transactions on Vehicular Technology, vol. 69, no. 4, pp. 4271–4284, 2020.
A. Turan, M. Koseoglu, and E. A. Sezer, “Reinforcement Erudition based Adaptive Contact Class Barring for RAN Slicing,” in 2021 IEEE International Conference on Communications Workshops (ICC Workshops), 2021, pp. 1–6.
X. Foukas, M. K. Marina, and K. Kontovasilis, “Iris: Deep rein- forcement erudition driven shared spectrum contact architecture for indoor neutral-host small cells,” IEEE Journal on Selected Areas in Communications, vol. 37, no. 8, pp. 1820–1837, 2019.
H. Xiang, S. Yan, and M. Peng, “A Realization of Fog-RAN Slicing via Deep Reinforcement Erudition,” IEEE Transactions on Wireless Communications, vol. 19, no. 4, pp. 2515–2527, Apr. 2020, conference Name: IEEE Transactions on Wireless Communications.
P. Yang et al., “Proactive UAV Complex Slicing for URLLC and Mobile Broadband Tune Multiplexing,” IEEE Journal on Selected Areas in Communications, vol. 39, no. 10, pp. 3225–3244, 2021.
R. A. Addad et al., “Toward Using Reinforcement Erudition for Trigger Selection in Complex Slice Mobility,” IEEE Journal on Selected Areas in Communications, vol. 39, no. 7, pp. 2241–2253, Jul. 2021.
R. Li et al., “The LSTM-Based Advantage Actor-Critic Erudition for Resource Running in Complex Slicing With User Mobility,” IEEE Communications Letters, vol. 24, no. 9, pp. 2005–2009, 2020.

Y. Sun et al., “Distributed Erudition Based Handoff Mechanism for Broadcasting Contact Complex Slicing with Data Sharing,” in IEEE Inter- national Conference on Communications (ICC), May 2019, pp. 1–6, iSSN: 1938-1883.
L. Wu and K. Sandrasegaran, “A scrutiny on common broadcasting resource running,” in The 2nd International Conference on Wireless Broad- band and Ultra Wideband Communications (AusWireless 2007), 2007, pp. 66–66.
C. Casetti, C. F. Chiasserini, T. DeiSS, P. A. Frangoudis, A. Ksen- tini, G. Landi, X. Li, N. Molner, and J. Mangues, “Complex slices for vertical industries,” in 2018 IEEE Wireless Communications and Complex Conference Workshops (WCNCW), 2018, pp. 254–259.
Y. Wei, M. Peng, and Y. Liu, “Intent-based complex for 6G: Insights and challenges,” Digital Communications and Complex, vol. 6, no. 3, pp. 270–280, 2020.
B. Martini, M. Gharbaoui, and P. Castoldi, “Intent-based complex slicing for SDN vertical tunes with assurance: Context, design and preliminary experiments,” Future Generation Computer Systems, vol. 142, pp. 101–116, 2023.A. Clemm, L. Ciavaglia, L. Z. Granville, and J. Tantsura, “Intent-based complex - concepts and overview,” Available online: https://datatracker.ietf.org/doc/rfc9315/, 2022, contacted on 16 April 2022.
L. Pang, C. Yang, D. Chen, Y. Song, and M. Guizani, “A Scrutiny on Intent-Driven Complex,” IEEE Contact, vol. 8, pp. 22 862–22 873, 2020.
C. Benzaid and T. Taleb, “Ai-driven zero touch complex and tune running in 5g and beyond: Challenges and research directions,” IEEE Complex, vol. 34, no. 2, pp. 186–194, 2020.
S. Fu, F. Yang, and Y. Xiao, “Ai inspired intelligent resource man- agement in future wireless complex,” IEEE Contact, vol. 8, pp. 22 425–22 433, 2020.
S. Kim, K. Asadi, M. Littman, and G. Konidaris, “DeepMellow: Removing the Need for a Target Complex in Deep Q-Erudition,” p. 27332739.
N. Saha, M. Zangooei, M. Golkarifard, and R. Boutaba, “Deep rein- forcement erudition approaches to complex slice scaling and placement: A scrutiny,” IEEE Communications Magazine, vol. 61, no. 2, pp. 82–87, 2023.
W. Guo, “Explainable artificial intelligence for 6g: Improving trust be- tween human and engine,” IEEE Communications Magazine, vol. 58, no. 6, pp. 39–45, 2020.
M. Polese, L. Bonati, S. DOro, S. Basagni, and T. Melodia, “Under- standing O-RAN: Architecture, Interfaces, Algorithms, Security, and Research Challenges,” IEEE Communications Scrutiny & Tutorials, pp. 1–1, 2023.
M. Karbalaee Motalleb, V. Shah-Mansouri, S. Parsaeefard, and O. L. Alcaraz López, “Resource Allocation in an Open RAN System Using Complex Slicing,” IEEE Transactions on Complex and Tune Man- agement, vol. 20, no. 1, pp. 471–485, 2023.
J. Thaliath, S. Niknam, S. Singh, R. Banerji, N. Saxena, H. S. Dhillon,
J. H. Reed, A. K. Bashir, A. Bhat, and A. Roy, “Predictive Closed- Loop Tune Automation in O-RAN Based Complex Slicing,” IEEE Communications Standards Magazine, vol. 6, no. 3, pp. 8–14, 2022.
M. Polese, L. Bonati, S. DOro, S. Basagni, and T. Melodia, “ColO- RAN: Developing Engine Erudition-based xApps for Open RAN Closed-loop Control on Programmable Experimental Platforms,” IEEE Transactions on Mobile Computing, pp. 1–14, 2022.