Customized Slice-aware Approach for Resource Allocation and Management of Slices in Beyond 5G Networks

Thiruvenkadam Srinivasan
Kunsan National University
Gunsan, South Korea
drstv.2011@gmail.com

In-Ho Ra
Kunsan National University
Gunsan, South Korea
ihra@kunsan.ac.kr

ABSTRACT

More personalized end-to-end services and new cloud-edge applications are to be expected from 6G networks than from current 5G networks. Assigning 6G mobile network resources while distributing resources across devices with diverse demands necessitates the use of network slicing (NS). Network function virtualization (NFV) and software-defined networking (SDN) technologies are driving the recent progress toward 6G NS. The problem of resource allocation in 6G is critical and warrants more investigation. The two most critical network characteristics, node mapping and link mapping, are the primary focus of virtual network resource allocation. In this work, slice-aware network embedding (SANE) strategy is adopted for resource allocation in 6G network. For node mapping, this work provides the slice specified node ranking technique and for link mapping, the k-shortest route algorithm. It is necessary to develop three sub-algorithms for the eMBB, uRLLC, and mMTC slices in the node mapping phase, each of which is responsible for creating the node ranking for the corresponding slice needs. For the purpose of verification, the suggested method is put to the test using a dynamic physical infrastructure. Data on average acceptance rates, cost-to-revenue and connection bandwidth use are utilized to assess the suggested strategy's success. To prove its efficacy, the findings of th is study are compared to other studies in the field.

KEYWORDS

5G, 6G, network slicing, resource allocation,

1 INTRODUCTION

Wireless communication is increasingly becoming an essential component of both the social infrastructure and people's daily lives. Mobile networks, from the first generation (1G) to the fifth generation (5G), have made significant contributions to the economic growth of the country. It is expected that the sixth generation (6G) mobile network will be commercially available by 2030, following in the footsteps of previous generations. 6G networks are expected to offer a wide range of services with varying quality of service (QoS) needs, such as multisensory extended reality, autonomous driving, and hologram video streaming [1]. Network slicing is a potential strategy for building several slices for different services on top of a shared physical network infrastructure. The lifespan of network slicing in 6G consists of three phases: creation, planning, and management, as shown in Fig. 1. While the centralized SDN controller is in responsible of the initial planning and preparation stages, the local SDN controllers are in charge of coordinating the operational stages [2].

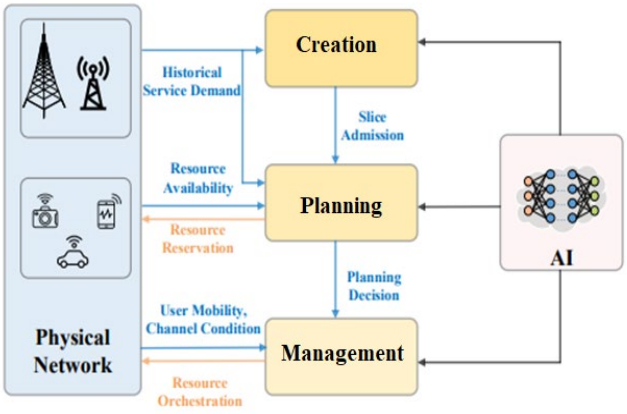

Figure 1: Life cycle of 6G Network Slicing

Due to the specific characteristics of 6G networks, advanced network slicing confronts numerous hurdles. However, the Space-air-ground integrated network (SAGIN) requires precise coordination of various network components in order to regulate slices across satellite, airborne, and terrestrial networks. To further complicate slice management, 6G networks must provide a wide variety of new services while meeting different and stringent QoS criteria [3]. Consequently, it is crucial to create intelligent slice management solutions for 6G networks. In addition, with the help of high-powered computing resources and cutting-edge AI approaches, pervasive intelligence is fostering a proliferation of emerging AI services that require new QoS characteristics like model correctness and learning s peed.

Network function virtualization (NFV) is recognized as a critical enabler for effective resource management and sharing in 6G networks [4]. An efficient resource computation approach is proposed for incorporation into the Resource Allocation-Service Function Chain-6G (RA-SFC-6G) [5]. The computation approach quantifies and reveals the resource capabilities of all physical nodes, becoming the foundation of the SFC embedding and scheduling challenge. A dynamic nested neural network is built, which adapts the nested learning model structure online to fulfill the training needs of dynamic resource allocation [6]. It presented an AI-driven

collaborative dynamic resource allocation (ACDRA) approach based on nested neural networks paired with Markov decision process training for huge IoT enabled by 6G.

TailoredSlice-6G [7] is an efficient approach for tailoring slice resource allocation in 6G networks. When a certain slice is received, the suggested algorithm first determines its type. After determining the slice type, the algorithm will choose the best sub algorithm for the customized resource allocation. Furthermore, this method is guaranteed to enable slice resource allocation in polynomial time. [8] proposes a RAN slicing technique with variable temporal and resource granularities to accommodate time-varying network conditions and diverse QoS needs. It is made comprised of an upper-level controller and a lower-level controller. The upper-level controller intends to improve service QoS performance by imposing slice configuration adaptation based on the dynamics of service traffic over a wide time scale. In [9], a joint network slicing and routing technique is provided, which combines the network management and control framework to provide fine-grained, real-time, and dynamic resource allocation. The suggested work uses Graph Convolutional Networks (GCN)-powered Multi-Task Deep Reinforcement Learning (DRL) to address the 6G resource allocation problem. In [10], security and QoS provisioning are implemented on a DAG-assisted Block chain-based SDN/NFV 5G Beyond and 6G with MEC environment. To begin, deep network slicing is presented for network virtualization by many slices and service kinds using GAN-SliceNet. A slice-based DAG block chain is built, and a proof of space (PoS) consensus technique is provided to overcome scalability difficulties. Then, context-aware user authentication is created, which makes use of three types of attributes: device, channel, and biometric.

A large number of technical algorithms for slice resource allocation have been described, but these algorithms all process the requested slices in a generic manner. "One algorithm serves all distinct types of slices," is how this approach is described in general. A short-term gain may be achieved, but a long-term loss of physical resources is certain. Furthermore, these algorithms can only be used in RANs or cores. As a result, all 6G networks will need a new resource allocation system. In this research, we offer a slice aware network embedding (SANE) strategy for allocating slices efficiently.

2 PROPOSED SYSTEM MODEL

2.1 Physical Infrastructure Model

The physical infrastructure (PI) of a network is modeled as an undirected graph $G_P = (N_P, E_P)$, where N_P indicates the set of nodes and E_P denotes the set of linkages in the physical structure. As with earlier research, this approach evaluates node capacity and security power level for each node. The capacity of a node is determined by the number of CPUs that are available. The total number of CPUs available in the i-th node is defined as, CPU_i. The security of i-th physical node is denoted by SPL_i, which represents the highest level of security that a physical node may provide to the NSR node. The primary parameter for connections linking nodes 'i' and 'j' is the link's bandwidth, BW_{ij}. The lengths of the various connections are supplied in order to discover the shortest route between any two nodes. For example, the variable $L_{i,j}$ stores the length of the line between the nodes 'i' and 'j'.

2.2 Network Slice Model

This study makes an assumption that each NSR requires the specified number of nodes, CPU capacity, bandwidth, security, and lifetime. Hence, NSR request is modeled as an undirected graph $G^{NSR} = (N_{NSR}, CPU_{NSR}, SPL_{NSR}, BW_{NSR}, LT_{NSR})$ where N_{NSR}, CPU_{NSR} and SPL_{NSR} are the set of nodes, CPU capacity and security power level of the associated NSR, respectively, BW_{NSR} is the set of bandwidth requirement of requested NSR nodes, and LT_{NSR} is the life time of the NSR. NSRs are assumed to be received at a single transmission time unit, one unit represents 1 min. During each interval, the physical infrastructure is used to allocate nodes and establish links for the NSR. Each time a new NSR is received, the physical infrastructure's available resources are updated depending on the preceding request's LT_{NSR}.

2.3 Deployment Strategy

It is easy to comprehend that the process of resource allocation is comprised of two primary embedding operations, namely node mapping and link mapping. For node mapping, the majority of previous studies [11-13] used the node ranking approach. The proposed SANE strategy employs the slice specific node ranking (NR) approach to improve resource allocation efficiency. Each sub-algorithm is responsible for preparing node ranking for the respective slice requirements such as eMBB, uRLLC, and mMTC. In this article, the well-known shortest path method [12] is used for finding the shortest path between two fixed nodes. The three node ranking approaches proposed for node mapping are detailed in the preceding subsections. Based on our knowledge, we decide to use the convenient resource value method (RVM) to score the node and link elements, which will serve as the basis for PI's sliced resource allocation. The RVM is a simple and widely used resource allocation model. The following expression can be used to calculate the resource value of any node 'i.'

$$RV(i) = CPU_i \, SPL_i \sum_{ij \in PI,NSR} BW_{ij} \tag{1}$$

where 'ij' refers to the link having one end node 'i'. More specifically, only the direct links of node 'i' has been considered.

2.3.1 NR for Resource Priority Type

In the eMBB communication situation, the Resource Priority Type has the potential to be implemented. Resource score values for all PI and NSR nodes may be readily accessed using equation (1). But the value of direct product indicates the local resource score. NSR or PI scores for each node need to be obtained in the most complete way possible. As a result, it is imperative that we build connections between all nodes in the network or slice. On the basis of the prior discussion [7], we are able to define two important categories of probabilities as follows: 1) neighboring probability ($P_{i,j}^{Nr}$) by using equation (2) and 2) jumping probability ($P_{i,j}^{jp}$) by using equation (3).

$$P_{i,j}^{Nr} = \frac{RV(i)}{\sum_{j \in Nr} RV(j)} \quad (2)$$

$$P_{i,j}^{jp} = \frac{RV(i)}{\sum_{j \in PI} RV(j)} \quad (3)$$

where Nr represents the specific node set, storing all neighboring nodes of 'j'. With respect to 'i' in the PI, its score value is formulated as follows, after 't' rounds of calculation:

$$RV(j)^{t+1} = \alpha \sum_{i \in Nr} P_{i,j}^{Nr} RV(i)^t + \alpha \sum_{i \in PI} P_{i,j}^{jp} RV(i)^t \quad (4)$$

We can determine the resources of all PI nodes using the iteration-based scoring approach above. Calculate the NSR nodes one by one by repeating the resource scoring algorithm. We then create two separate sets to hold the PI and NSR nodes. Resources are ranked in decreasing order of their worth. After that, NSR's resources will be allocated. With the greatest NSR resource score value, node 'i' is prioritized for processing. Priority is given to physical node 'I', which in PI has the greatest resource score value. It is necessary to keep checking to see whether I's network, time, and security requirements can be met if 'I' has plenty of CPU, storage, and capacity. In this case, we may choose 'I' to fit 'i', and do the same procedure for the remaining virtual nodes in NSR. All nodes' resource, function, time and security requirements will be met before we allocate link resources for NSR. Only bandwidth demand is addressed in this article when it comes to the distribution of NSR link resources.

2.3.2 NR for More Slice Element Type

More Slice Element Type might be beneficial for network slices that have a greater connectivity, such as mMTC. This study has developed a subalgorithm that enables us to identify the physical network components of the More Slice Element Type that have a better connection and more resources. The study establishes a new resource measure that is based on direct links and the resources supplied to nodes in the network. This resource measure, which is called a "direct-link block," is the result of this development (DLB).We formulate the DLB of certain one node 'i' of PI with 'N' number of direct links in expression (5).

$$DLB(i) = (CPU_i + SPL_i)\ N_i \sum_{ij \in PI} BW_{ij} \quad (5)$$

In place of RV, the DLB value of nodes may be substituted into equation (4) to get the stable connectivity value of nodes for this slice type. When it comes to the distribution of node resources, the allocation approach follows the same pattern as the distribution strategy for resource priority types.

2.3.3 NR for Time Sensitive Priority Type

It is possible to employ this slice type in the uRLLC communication scenario in order to provide the particular time-sensitive service. The timing requirements for these kinds of slices are rather stringent. In this piece, the main focus is given to the node processing delay metric (P). The following equation is the one that should be used to calculate the value of the time block corresponding to node 'i':

$$TB(i) = (CPU_i + SPL_i)\ P_i \sum_{ij \in PI} BW_{ij} \quad (6)$$

The allocation approach for node resources is the same as the allocation strategy for the previous two kinds.

2.4 Performance Metrics

The resource efficiency and the acceptance rate are two essential measurements that are used to examine the effectiveness of slice provisioning.

2.4.1 Resource Efficiency.

This metric is measured by calculating the achieved revenue and the investment cost made for providing the physical infrastructure. The revenue can be determined from the CPU capacity of nodes and link bandwidth requested for the NSRs. The investment cost is estimated from the physical infrastructure for the cause. The expression for calculating the resource efficiency (RE) is given by,

$$P_{RE} = \frac{cResources}{aResources} \quad (7)$$

where, *cResources* and *aResources* are consumed and available resources of physical infrastructure, respectively.

2.4.2 Acceptance Ratio

This metric provides the direct measurement of the adapted network slicing technique on the given physical infrastructure. It is measured by taking the ratio between successfully completed NSRs and unsuccessful NSRs for a given time 'T_{max}' which is expressed as,

$$P_{AR} = \frac{sNSR}{tNSR} \quad (8)$$

where, *sNSR* and *tNSR* are served and total NSR, respectively.

2.5 Resource Allocation and Performance Assessment

The sequence of events that take place during the entirety of the proposed SANE strategy is depicted in Fig. 2. These events allot resources for NSRs for the longest possible amount of time (Tmax). Node mapping begins with a suitable node ranking technique for both PI and NSR that is selected from (4), (5), and (6) during the time interval when the NSR is received.

3 RESULTS AND DISCUSSION

The proposed work has been executed on physical infrastructure that has three distinct resource capacities. In addition, this is carried

out based on the varied amounts of NSRs received within the allotted time range. Table 1 details the physical infrastructure and NSR implementation aspects that are examined.

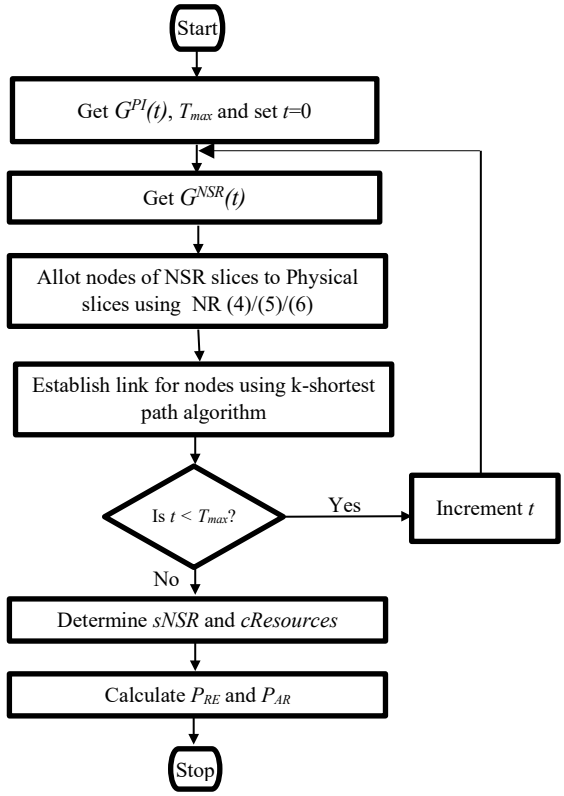

Figure 2: Sequence for resource allocation

Table 1: Test Case Parameters

Definitions	Descriptions	Range
N_{PI}	Number of physical nodes	100,200,300
CPU_{PI}	The distribution of CPU for each node in unit	U[30,60]
SL_{PI}	The distribution of available security level of a node in real number	(0-1)
BW_{PI}	The distribution of bandwidth of each links in unit	U[30,60]
T_{NSR}	The total number of NSRs arrived in the time frame	U[5,35]
N_{NSR}	The distribution of nodes for each NSR	U[10,30]
CPU_{NSR}	The distribution of CPU requirement	U[5,25]
BW_{NSR}	The distribution of bandwidth requirement	U[5,25]
SPL_{NSR}	The distribution of required security level of a node in real number	(0-0.5)
LT_{NSR}	The time duration of each NSR	T[10,40]

According to dynamic provisioning, the request lifetimes are randomly selected from the range indicated in Table 1. In addition, the resources used by embedded requests in the physical infrastructure are released after the embedded request's lifetime has expired, so preserving the resources for future requests. Through resource efficiency and acceptance rate, the efficiency of the planned works is proved under varying conditions of PI and NSR. Comparing the acquired results with those found in the literature, notably NNR[11], CN[12], and VIKOR[13], indicates the usefulness of the suggested approach. All of the suggested methods are assumed to have access to the dynamic provisioning service for network requests.

Resource efficiency may be quantified in terms of the amount of time and money spent on a physical resources. Nodes in the PI and the amount of requests they get affect the value, which changes depending on the number of nodes and requests. As shown in Figure 3, the suggested method's resource efficiency might vary depending on the situation. Resource efficiency equation shows that more nodes in a network results in a shorter overall shortest path, which in turn leads to better resource use as a consequence. As the number of NSRs grows, so does the need for more nodes with sufficient CPU and bandwidth. Fig. 3 shows that, when implemented to a 100-node infrastructure, the suggested technique has a resource efficiency of 0.783, 0.765, and 0.753 for NRs 10, 20, and 30. As the number of nodes rises to 200 and 300, the resource's efficiency climbs to 0.8 under 5 NSRs, as demonstrated in Figure 3 for a physical resource with 300 nodes.

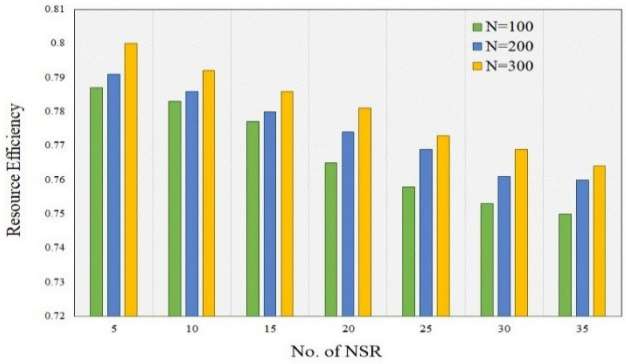

Figure 3: Resource Efficiency under different PI and NSR conditions

NSRs ranging from 5 to 35 are used during the course of this experiment in order to determine the general acceptance rate of the suggested methodology. Fig. 4 presents the outcomes that were attained as a consequence of the algorithm being carried out effectively in accordance with a wide range of criteria that relate to the physical infrastructure. Even if there are more physical nodes being added to the network, the acceptance ratio will fall as the total number of requests continues to rise. As a result of their being more nodes available, a greater percentage of the whole population has come around to adopting it. The acceptance ratio continues to rise in tandem with the expansion of the number of nodes that make up the physical infrastructure. According to Fig. 4, the suggested technique reaches an acceptance rate of 100% after receiving 5 NRs, despite the fact that the number of nodes in its PI does not change. In addition, looking at Fig. 4 reveals that the acceptance rate for the service request for 35 NRs in PI with 100 nodes is a dismal 0.4%.

An algorithm's performance against that of other algorithms has been evaluated using PI, which has 300 nodes to handle a broad range of network requests while also accounting for factors such as resource efficiency and user acceptance rate. It's easy to see how the different algorithms stack up in Fig. 5, which categorizes them according to their performance. Fig. 5 shows that when all methods are combined, the CN algorithm has the lowest throughput. VIKOR, NNR and SANE all produced better results than CN. NSRs ranging from 5 to 35 are used to assess the suggested technique of determining the acceptance rate. It is decided in advance that the NSRs' life durations will be chosen at random from a range of possibilities.

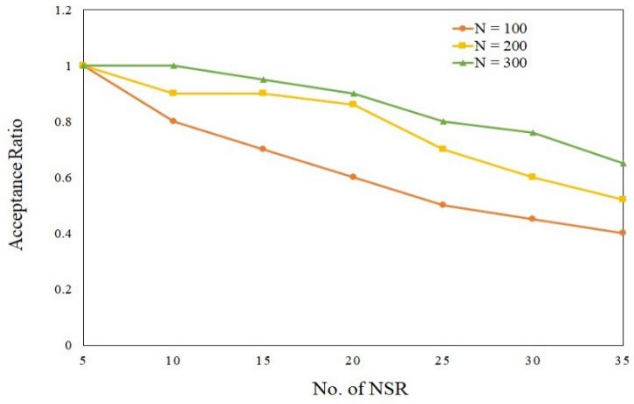

Figure 4: Acceptance ratio under different PI and NSR conditions

Fig. 6 shows the results produced after the algorithms were successfully run on a real infrastructure with 300 nodes. When the total amount of NSR points is raised, the acceptance ratio decreases. NSR acceptance rates tend to rise as the lifetime of the species shortens. The suggested approach can achieve an acceptance ratio of up to 0.98, whereas 35 NSRs can achieve an acceptance ratio of up to 0.7. Due to its performance being on par with the SANE method, the NNR algorithm is placed second. For 35 NRs under accessible nodes, the CN and VIKOR techniques have the lowest acceptance rates of 0.68 and 0.66, respectively. Each node's count is included into this calculation.

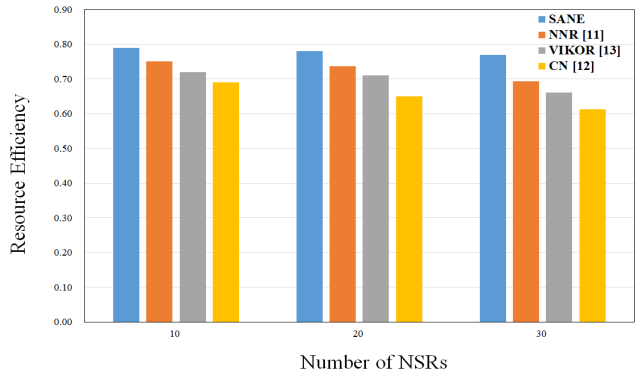

Figure 5: Comparison of results for resource efficiency

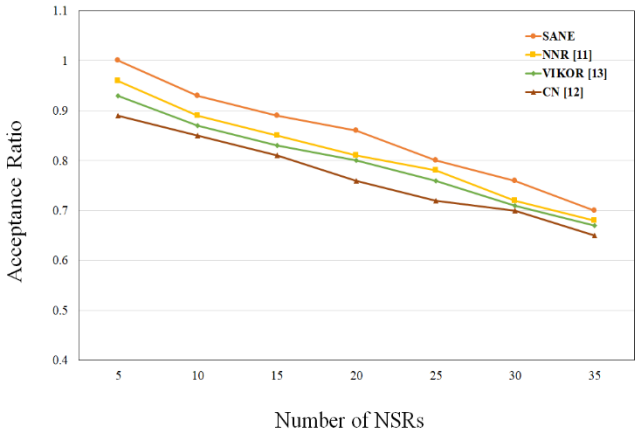

Figure 6: Comparison of results with acceptance rate

4 CONCLUSION

During the course of this study, a solution strategy for effective network slicing in the context of a 6G mobile network environment was established. The deployment strategy was modeled in its entirety, including the NSRs and the physical infrastructure. The suggested technique takes into consideration the most important aspects of network slicing, namely the isolation of slices and their administration. The proposed SANE technique employs slice-aware node ranking and the k-shortest route algorithm for node allocation and connection construction, respectively. The suggested SANE technique leverages the slice-specific node ranking method to increase the effectiveness of resource allocation. Each sub-algorithm is responsible for producing the node ranking for the specific slice needs, including eMBB, uRLLC, and mMTC. The suggested solutions were assessed across a broad range of network operating situations, each with its own PI and NSR values. To determine the efficacy of network slicing, we examined resource efficiency and acceptance rate. By incorporating limits into the resource allocation procedure, issues about the NSRs' limited SLA and security have been resolved. The suggested work may be developed to include a higher degree of SLA and security considerations. In addition, it could be enhanced to account for the mobility of user equipment and its energy management.

ACKNOWLEDGMENTS

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2021R1A2C2014333).

REFERENCES

[1] Y. Huang *et al*., "6G mobile network requirements and technical feasibility study," in *China Communications*, vol. 19, no. 6, pp. 123-136, June 2022, doi: 10.23919/JCC.2022.06.010.

[2] Wu, Wen & Zhou, Conghao & Li, Mushu & Wu, Huaqing & Zhou, Haibo & Zhang, Ning & Xuemin, & Shen, Xuemin & Zhuang, Weihua. (2021). AI-Native Network Slicing for 6G Networks.

[3] S. Kukliński, L. Tomaszewski, R. Kołakowski and P. Chemouil, "6G-LEGO: A framework for 6G network slices," in Journal of Communications and Networks, vol. 23, no. 6, pp. 442-453, Dec. 2021, doi: 10.23919/JCN.2021.000025.

[4] X. You, C. Wang et al., “Towards 6G wireless communication networks: Vision, enabling technologies, and new paradigm shifts,” Sci. China Inf. Sci., vol. 64, no. 1, Jan. 2021, Art no. 110301

[5] H. Cao et al., "Resource-Ability Assisted Service Function Chain Embedding and Scheduling for 6G Networks With Virtualization," in IEEE Transactions on Vehicular Technology, vol. 70, no. 4, pp. 3846-3859, April 2021, doi: 10.1109/TVT.2021.3065967.

[6] K. Lin, Y. Li, Q. Zhang and G. Fortino, "AI-Driven Collaborative Resource Allocation for Task Execution in 6G-Enabled Massive IoT," in IEEE Internet of Things Journal, vol. 8, no. 7, pp. 5264-5273, 1 April1, 2021, doi: 10.1109/JIOT.2021.3051031.

[7] H. Cao et al., "Toward Tailored Resource Allocation of Slices in 6G Networks With Softwarization and Virtualization," in IEEE Internet of Things Journal, vol. 9, no. 9, pp. 6623-6637, 1 May1, 2022, doi: 10.1109/JIOT.2021.3111644.

[8] J. Mei, X. Wang, K. Zheng, G. Boudreau, A. B. Sediq and H. Abou-Zeid, "Intelligent Radio Access Network Slicing for Service Provisioning in 6G: A Hierarchical Deep Reinforcement Learning Approach," in IEEE Transactions on Communications, vol. 69, no. 9, pp. 6063-6078, Sept. 2021, doi: 10.1109/TCOMM.2021.3090423.

[9] T. Dong et al., "Intelligent Joint Network Slicing and Routing via GCN-Powered Multi-Task Deep Reinforcement Learning," in IEEE Transactions on Cognitive Communications and Networking, vol. 8, no. 2, pp. 1269-1286, June 2022, doi: 10.1109/TCCN.2021.3136221.

[10] I. H. Abdulqadder and S. Zhou, "SliceBlock: Context-aware Authentication Handover and Secure Network Slicing using DAG-Blockchain in Edge-assisted SDN/NFV-6G Environment," in IEEE Internet of Things Journal, doi: 10.1109/JIOT.2022.3161838.

[11] Cao, H.; Yang, L.; Zhu, H. Novel node-ranking approach and multiple topology attributes-based embedding algorithm for single-domain virtual network embedding. IEEE Internet of Things Journal 2017, 5, 108–120.

[12] W. Guan, X. Wen, L. Wang, Z. Lu and Y. Shen, "A Service-Oriented Deployment Policy of End-to-End Network Slicing Based on Complex Network Theory," IEEE Access, vol. 6, pp. 19691-19701, 2018, DOI: 10.1109/ACCESS.2018.2822398, [online].

[13] X. Li, C. Guo, L. Gupta and R. Jain, "Efficient and Secure 5G Core Network Slice Provisioning Based on VIKOR Approach," IEEE Access, vol. 7, pp. 150517-150529, 2019, DOI: 10.1109/ACCESS.2019.2947454, [online].