2019 International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)

LRA-3C: Learning Based Resource Allocation for Communication-Computing-Caching Systems

(Invited Paper)

Ge Wang†, Li Wang†, Jianbin Chuan†, Wenjing Xie†, Hongming Zhang‡, and Aiguo Fei‡
† School of Electronic Engineering, Beijing University of Posts and Telecommunications, Beijing, P.R. China.
‡ School of Software Engineering, Beijing University of Posts and Telecommunications, Beijing, P.R. China.

***Abstract*—The diversified usage scenarios of 5G networks require to allocate communication-computing-caching (3C) multi-dimensional resources efficiently according to different service requirement in terms of latency, bandwidth, and connectivity. In this paper, a network slicing (NS) architecture based mobile edge computing (MEC) and software-defined network (SDN) technologies are proposed to support flexible 3C resource allocation for improving the service of three 5G typical usage scenarios, namely enhanced mobile broadband (eMBB), ultra-reliable and low latency communications (URLLC) and massive machine-type communications (mMTC). Furthermore, a neural network (NN) based 3C resource allocation algorithm is designed to provide resource allocation decision for the NS architecture. With the aid of data pre-processing techniques, fast resource allocation decisions can be made by the NN aided NS architecture. Meanwhile, the performance of the proposed NS architecture is investigated in a testbed environment. Experimental results obtained from the testbed demonstrate that the system performance of the NN aided NS architecture can be improved by function fitting based pre-processing approaches. Moreover, an accurate classification performance can be obtained by the proposed architecture for facilitating 3C resource allocations.**

***Index Terms*—Network slicing, 5G, neural networks, MEC, SDN**

I. Introduction

With the explosive growth of wireless devices and the emergence of numerous applications brought by the rapid development of society, it is challenging to design an efficient wireless network architecture for addressing issues, such as massive machine-type communication demands of Internet of Things (IoT), low latency and high data rate requirements of virtual reality, etc. For this reason, service-oriented 5G network is developed to provide better user experience and higher data rate [1]. Specifically, three scenarios, namely, enhanced mobile broadband (eMBB), ultra-reliable and low latency communications (URLLC) and massive machine-type communications (mMTC), have been proposed by ITU-R for providing communication services with broadband, low latency, high reliability, and large connection, respectively [2].

In 5G, a key issue is how to make the full use of existing resources and universal infrastructures for providing customized services, since the improvement of network capabilities by deploying more infrastructures is not only cost inefficient, but also unable to meet the demands of the fast growth of mobile traffic [3]. In order to tackle this issue, software-defined networking (SDN) with high flexibility in resource management and high programmability in service orchestration is conceived [4]. Meanwhile, network slicing (NS) is capable of virtualizing resources. Hence, different resource configurations and infrastructure settings can be facilitated for different services [5]. In general, NS is used to construct end-to-end logical networks (i.e. network slices) on demand for providing customized network services. It is capable of providing flexible resource allocation solutions for the dynamic network services on a uniform infrastructure. Moreover, different network slices are isolated to ensure the security and robustness of the network. In contrast to cloud computing, MEC [12] is capable of providing computing and caching services at the edge of networks. It has the advantages of low latency, high data rate, as well as low energy cost. In 5G, communication, computing, and caching (3C) resources are required to be deployed more flexible, where a huge 3C resource pool exists in the network [19]. However, it cannot be ignored that the explosive growth of network traffic in 5G yields an increased demand for 3C resources [18]. Hence, it is significant to optimize resource utilization by effectively allocating 3C resources according to various service types. Moreover, in comparison to 1C/2C resource allocation, higher system performance and better user experience can be achieved by efficient 3C resource allocation schemes [17].

Due to the limitation of network infrastructure, it is important to achieve efficient utilization of networks by allocating resources to different network slices according to their service characteristics. To this end, considerable research attention has been paid. In order to maximize energy efficiency and meet proposed Quality of Service (QoS), Alqerm *et al.* proposed centralized and decentralized enhanced online learning approaches for downlink multi-tier heterogeneous networks [6]. In [7], Hussien proposed a novel resource allocation scheme by minimizing the bandwidth of periodic machine-to-machine traffic subjecting to diverse QoS requirements of machine-type communication devices as well as allowing the admission of new machine-type communication devices in a flexible manner. The authors of [8] proposed a resource allocation scheme, which is able to effectively allocate network resources to network slices. Raza *et al.* showed a heuristic slice admission strategy based on big data analytics predictions in [23]. In [10], Halabian *et al.* proposed a distributed scheme to solve the resource allocation problem based on the notion of dominant resource fairness with non-collaborative slices. In [17] Tang proposed a generalized

This work was supported in part by the National Natural Science Foundation of China under Grant 61571056, 61871416, in part by the Beijing Science and Technology Nova Program under Grant xx2018083, in part by the Fundamental Research Funds for the Central Universities under Grant 2018XKJC03, and in part by the Beijing Municipal Natural Science Foundation under Grant L172010.

L. Wang (*Corresponding author*) is with the Key Laboratory of the Universal Wireless Communications, Ministry of Education, Beijing 100876, China (e-mail: liwang@bupt.edu.cn).

978-1-7281-2980-8/19/$31.00 ©2019 IEEE

DOI 10.1109/iThings/GreenCom/CPSCom/SmartData.2019.00150

3C resource sharing framework and an algorithm-based on linear programming, in order to provide an empirically close-to-optimal solution with a reduced computation time. However, in practice, the aforementioned work might be unable to be adapted to the blossoming services. Furthermore, it is difficult for traditional methods to be applied to the existing infrastructures. These issues can be addressed by a popular machine learning method, which is known as the neural network (NN). NN is an information processing system based on the structure and function of brain neural network [11], which has been applied to some applications in the field of communications. In [14] a NN architecture is specifically designed to facilitate the spatial learning of geographical locations of interfering or interfered nodes. Considering the trust-based MSNs with MEC, caching and D2D, He *et al.* proposed a big data deep reinforcement learning approach to make a decision for optimally allocating 3C resources [23].

Against the above background, our contributions of this paper are summarized as follows.

- *We propose a novel NS architecture which supports 3C resource allocation. In our architecture, MEC is employed to provide flexible computing and caching resources near mobile users, as well as to make the full use of existing resources for reducing network costs. Furthermore, SDN is used to provide a flexible resource management for our proposed NS architecture.*
- *We propose an innovative resource allocation scheme by using NN, where multidimensional network resources, including communication, computing, and caching, are trained for obtaining an exclusive NN model. In this way, effective resource allocation strategies can be provided for our proposed NS controller.*
- *Finally, the performance of the proposed NS architecture is investigated in a testbed environment. Specifically, the system performance is evaluated in the scenarios of eMBB, URLLC, and mMTC, showing that our proposed architecture is capable of classifying service types accurately and allocating 3C resources effectively to improve the network performance and user experience.*

II. Design of NS Architecture for 3C Resource

In this section, we will introduce the NS architecture designed based on MEC and SDN technologies, which is oriented to typical 5G usage scenarios and assigns different network slices to users with different types of services which result in different service demands and different channel conditions. We can ensure efficient utilization of resources, cost reduction and convenience of deployment, so as to achieve significant improvement of user experience.

A. System Design

As shown in Fig. 1 where 3C resource can be collaboratively allocated. According to dynamic service requests, different 3C resource allocation strategies are made by the Controller and the Agent. Controller makes 3C resource allocation decisions for service requests. Agent executives decisions from Controller. As shown in Fig. 1, 3C resources can be virtualized and divided into isolate parts using SDN. Hence, NS technique can be adopted for resource allocation. Moreover, with the aid of SDN, network topology, resource distribution, resource status, channel condition, network delay, etc, can be obtained for deploying different applications with specific service demands. As a result, it is possible to obtain optimized resource allocation strategies for different service requirement. On the top of Fig. 1, we consider three cases, which corresponds to three service types in 5G.

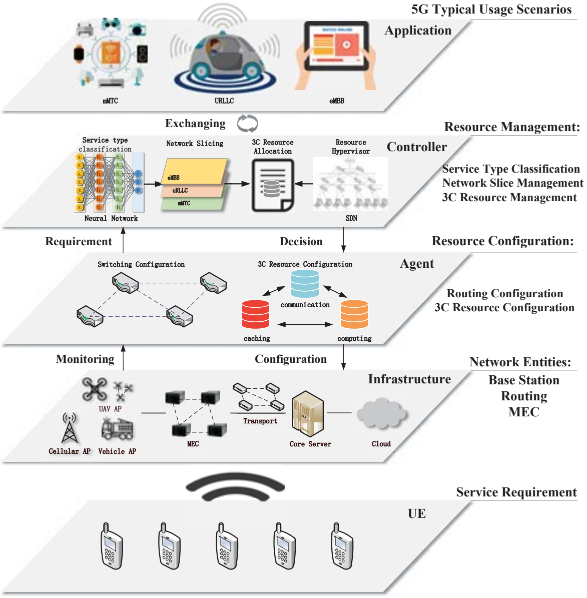


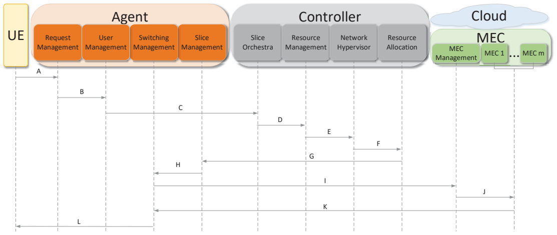
Fig. 1: System scenario of NS.

B. System Architecture

As shown in Fig. 2, our proposed system is consisted of five parts, which are user equipment (UE), MEC, Controller, Agent and Cloud. The UE is both the sender and the receiver of service requirements. The Agent is responsible for delivering the UE's service requirements to the Controller, as well as receiving and executing the Controller's decision. As the intermediary between Controller and UEs, the Agent is designed with the following four function modules: Slice Management, User Management, Request Management, and Switching Management. The Slice Management is used to allocate network resources to network slices based on resource allocation strategies issued by Controller. The User Management provides network access services and channel status. The Request Management is responsible for managing user requests and forwarding service demand information to the Controller. The Switching Management is responsible for establishing the forwarding routing table.

In general, service tasks are first forwarded to MEC by Agent. Then, MEC Management helps to connect with MEC server providing the service capability of computing and caching. By contrast, the Cloud is responsible for computing tasks that are out of the capability of MEC. For simplify, we assume that there is only one Cloud in our architecture. As shown on the top of Fig. 2, our Controller is designed for network slices as well as 3C resource allocations. The Controller contains the following functional modules: Network Hypervisor, Resource Management, Slice Orchestra, and Resource Allocation. The Network Hypervisor is used to monitor and obtain the network topology. The Resource Management is responsible for monitoring resource status of each MEC and obtaining the available resources. The Slice Orchestra is used to obtain the service demands from the Agents based on different user applications and creates network slices for them. Resource Allocation is responsible for obtaining the resource allocation strategies.

Specifically, Controller monitors the network topology, the distribution of multidimensional resources, and their usage. At the same time, it manages the creation and logout of the network slice realizing the exclusive matching between the service task and the network slice. Hence, SDN controller is the concrete implementation of the SDN architecture. The separation of control plane and data plane is one of basic principles in SDN technology. Following this principle, we design the Controller in our architecture for managing 3C resources and transmitting information in the control plane. Centralized architecture is adopted to generate the topological graph of the network and the communication-computing-caching multidimensional resources allocation.



A: send service requirement
B: deliver service requirement
C: forward service requirement
D: create NS based on service type
E: deliver resource distribution
F: deliver network topology
G: forward resource allocation decisions
H: deliver resource allocation decisions
I: connect computing and cache resource, forward service task
J: connect MEC, complete service task and send service result
K: forward service result
L: deliver service result
M: delete service requirement
N: log out NS

Fig. 2: Proposed NS based flowchart for 3C resource allocation.

C. 3C Resource Allocation in the Proposed Architecture

As shown in Fig. 2, service request is first sent to the Request Management of Agents from UE. Then, this service request is stored in Request Management, and delivered to User Management. As shown in Fig. 2, User Management forwards the service request to Slice Orchestra in the Controller, where a dedicated network slice is created for each service task and each service task is identified. The results are delivered to Resource Management, where the corresponding resource distribution are sent to Network Hypervisor. In the Network Hypervisor, network topology is obtained. Next, resource allocation decision is made and sent to Slice Management module of Agents. Resource allocation decision is forwarded to Switching Management as seen in Fig. 2.

We notice that Resource Allocation in the Controller provides 3C resource allocation for the service. It is important to efficiently allocate 3C resources to ensure service quality. In the next section, we propose an NN to achieve the 3C resource allocation according to the channel status and service demands.

III. Neural Network Based Multidimensional Resource Allocation Scheme

In this section, the structure of the neural network which provides the resource allocation decision for the Controller will be described in detail. At the same time, we also present the details of the NN based multidimensional resource allocation algorithm that provides the decision for the Controller.

A. Neural Network Based Service Decision for NS

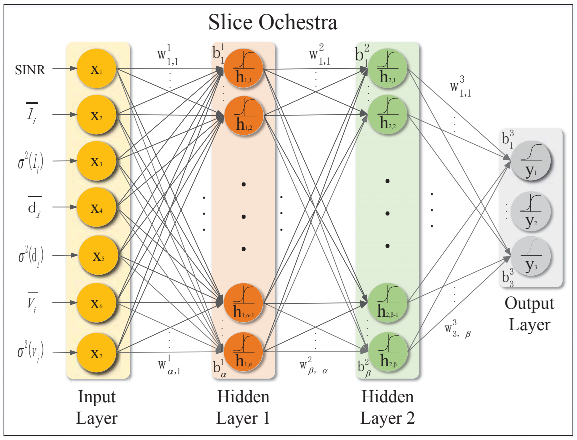


Fig. 3: Neural network model to identify the service type.

As mentioned in Section II, during a service for a UE, a service request is forwarded to the Controller by the Agent. We propose an NN to identify the service type of a user request based on the corresponding channel status and service demands. In our proposed NN, service type is identified according to signal-to-interference and noise ratio (SINR) of the UE, the length of a packet, service delay, as well as data rate. In general, several packets are generated and transmitted in this procedure. In order to refine the characteristics of the service, the mean and the variance of the length of these packets are calculated. Let us denote the corresponding mean and the corresponding variance as $\overline{l_i}$ and $\sigma^2(l_i)$ respectively, where the subscript i represents the ith service type of the UE requested. Similarly, the mean of service delay $\overline{d_i}$ and the variance of service delay $\sigma^2(d_i)$ are calculated. Finally, the mean and the variance of data rates, which are denoted as $\overline{v_i}$, and $\sigma^2(v_i)$, are obtained. The proposed NN can be expressed as a mapping from input layer to output layer, which can be expressed as

$$\boldsymbol{y} = f(SINR, \overline{l_i}, \sigma^2(l_i), \overline{d_i}, \sigma^2(d_i), \overline{v_i}, \sigma^2(v_i)), \tag{1}$$

where the above-mentioned parameters are fed into our proposed NN as inputs. As seen in Fig. 3, there are two hidden layers in our proposed NN, where the first hidden layer contains α neurons and the second hidden layer contains β neurons. Specifically, the function of each hidden layer can be expressed as

$$h_{m,k} = f\left(\sum_j w^m_{k,j} h_{(m-1),j} + b^m_k\right), \tag{2}$$

where $h_{m,k}$ denotes the output of the $k\text{-}th$ neuron in $m\text{-}th$ hidden layer. Here, we have $h_{0,j} = x_j$ for $m = 1$, representing $j\text{-}th$ input of NN. In (2), $w_{k,j}^m$ is the weight of the $j\text{-}th$ neuron in $(m-1)\text{-}th$ hidden layer to the $k\text{-}th$ neuron in the $m\text{-}th$ hidden layer. Meanwhile, b_k^m is the bias of the $k\text{-}th$ neuron at $m\text{-}th$ hidden layer. Moreover, we use $softmax$ function [15] as the activation function,

$$softmax_i = \frac{e^{a_i}}{\sum_d e^{a_d}}, i = 1, 2, ..., n, \tag{3}$$

Finally, in the output layer, we obtain $\boldsymbol{y} = [y_1, ..., y_n]$, where $y_i \epsilon \{0, 1\}$ is defined as a service selection indicator for $i = 1, 2, ..., n$. In Fig. 3, $\boldsymbol{y}$ is used to identify three service types, which are eMBB, URLLC, mMTC with different resource configurations. Accordingly, each service type corresponds to a specified NS which can be denoted as $S_j,\ j = 1, 2, 3$. Here, the 3C resource allocation of S_j can be denoted as communication θ_j, computing φ_j, and caching ρ_j.

TABLE I

Service	Indicator $\boldsymbol{y}$	3C Resources
eMBB	[1, 0, 0]	$\mathcal{S}_1 = \{\theta_1, \varphi_1, \rho_1\}$
URLLC	[0, 1, 0]	$\mathcal{S}_2 = \{\theta_2, \varphi_2, \rho_2\}$
mMTC	[0, 0, 1]	$\mathcal{S}_3 = \{\theta_3, \varphi_3, \rho_3\}$

As mentioned above, in Slice Orchestra, service type is identified, which is a classification problem . Let us define $h_{3,i}$, which is obtained from (3), as the probability at the output layer. Let us define $\hat{\boldsymbol{y}} = [\hat{y}_1, ..., \hat{y}_n]$ as the label of training data. Hence, the objective function is given by [16] the likelihood of the $i\text{-}th$ service type obtained by NN,

$$J(\boldsymbol{w}) = -\sum_{i=1}^{n} \left[\hat{y}_i log h_{3,i} + (1 - \hat{y}_i) \, log\,(1 - h_{3,i})\right], \tag{4}$$

where $h_{3,i}$ is obtained from (3) with $0 < h_{3,i} \leq 1$. Finally, NN is trained to obtain the model weights in w by solving the optimization problem of

$$\min_{w} J(\boldsymbol{w}), \tag{5}$$

which can be solved by the well-known gradient descent algorithms [16].

During the NN training stage, since the size of training samples is large, a pre-process method is used, where data fitting methods are employed for obtaining the mean and the variance of the input parameters of NN. Then, a number of training samples are randomly selected as a group to train our NN. Finally, NN is trained off-line to support on-line 3C resource allocations. In order to provide high accuracy in a dynamically online environment, NN will be updated in time. The detailed off-line training algorithm of our NN is summarized in Algorithm 1.

B. Neural Network Based Multidimensional Resource Allocation Algorithm

In the Controller of our system architecture mentioned above, we obtain the service request of UE by Agent, and pre-process it as the input of the trained NN to classify the service type of the request denoted as $\mathcal{S}_r$. Then, we calculate the 3C resources allocation decision $\mathcal{S}_a$ according to $\mathcal{S}_r$, θ_{free}, φ_{free}, and ρ_{free} Here θ_{free}, φ_{free}, and ρ_{free} are the free and available resource of MEC corresponding to communication, computing, and caching, respectively. Only if $(\theta_r \leq \theta_{free}) \cap (\varphi_r \leq \varphi_{free}) \cap (\rho_r \leq \rho_{free})$ can MEC finish the service. When MECs can not finish the task, Controller will deliver the task to the Cloud. The detailed steps for resource allocation are summarized in Algorithm 2.

Algorithm 1: Neural Network Model Training Algorithm for Service Type Classification

```
Input: Original data, Number of service types N,
       Limitation of loss $\varepsilon$, Activation function $f$
Output: $\boldsymbol{w}$, $\boldsymbol{b}$
for $k$=1:N do
    Step 1: Pre-processing
    for $i$=1 to $M$ do
        Allocating resource according $\mathcal{S}_i$;
        Sampling dataset on test platform;
        Function fitting;
        Calculating $\overline{l_i}$, $\sigma^2(l_i)$, $\overline{d_i}$, $\sigma^2(d_i)$, $\overline{v_i}$ , $\sigma^2(v_i)$;
    end
end
Step 2: NN Training
Initialize: $\boldsymbol{w} = \left[w_{k,j}^m\right] \epsilon \mathcal{R}^{N \times JK}$, $\boldsymbol{b} = \left[b_k^m\right] \epsilon \mathcal{R}^{N \times K}$;
repeat
    $h_{1,k} = f\left(\sum_j w_{k,j}^1 x_j + b_k^1\right)$;
    $h_{2,k} = f\left(\sum_j w_{k,j}^2 h_{1,j} + b_k^2\right)$;
    $h_{3,k} = f\left(\sum_j w_{k,j}^3 h_{2,j} + b_k^3\right)$;
    Calculate $J(\boldsymbol{w})$ by using (4);
    Update $\boldsymbol{w}$, $\boldsymbol{b}$ via gradient descent method;
until $J(\boldsymbol{w}) < \varepsilon$;
```

IV. Implementation and Performance Analysis

In this section, we introduce the technical details of our implementation platform, which is used to evaluate the performance of our system. At the same time, we introduce our experiments for testing the resource allocation scheme and present the accuracy of neural network in real communication environment.

A. Implementation Platform

Data samples are collected based on the platform shown in Fig. 5. Commercial mobile phones are used as UEs in our testbed. Moreover, Baicell HaloB (http://www.baicells.com/en/) is employed as a base station (BS) to provide wireless network access in 2.4GHz. A micro-computer (Intel NUC8i7HNK4, RAM:16G, SSD:1T) is deployed as MEC, and a desktop computer (HP Battle99-53, i7-8750H, RAM:16G, SSD:512G) is used to provide NS. Here, our Controller is built on the Floodlight SDN controller shown in Fig. 5. MECs are implemented with the aid of virtual machine in the micro-computer for providing different computing and caching capabilities on

Algorithm 2: Neural Network Based 3C Resource Allocation Algorithm

```
Input: w, b, service request, $\theta_{free}, \varphi_{free}, \rho_{free}$, Number
       of MECs $m$
Output: 3C resource allocation $S_a$, allocated MEC $M$
Step 1: Pre-processing
Preprocess service request;
Calculate SINR, $\overline{l_r}$, $\sigma^2(l_r)$, $\overline{d_r}$, $\sigma^2(d_r)$, $\overline{v_r}$, $\sigma^2(v_r)$;
Calculate $\mathcal{S}_r$ using the trained neural network model;
Obtain $\theta_r$, $\varphi_r$, $\rho_r$;
Step 2: 3C Resource Allocation
$\mathcal{S}_a = null$;
for i=1 to m do
    if $(\theta_r \le \theta_{free}) \cap (\varphi_r \le \varphi_{free}) \cap (\rho_r \le \rho_{free})$ then
        $\theta_a = \theta_r, \varphi_a = \varphi_r, \rho_a = \rho_r$;
        $\mathcal{S}_a = \{\theta_a, \varphi_a, \rho_a\}$;
        $M = m$;
    end
end
if $\mathcal{S}_a == null$ then
    Deliver service request to the Cloud;
end
```

different network nodes. In the micro-computer, data packets and 3C resources are forwarding to the Controller by Open vSwitch in the Agent. Finally, data samples are obtained and preprocessed to form a dataset.

The statistical characteristics of the packet length of the three service types is shown in Fig. 4. It is obvious that the number of packets decreases as the length of packet increases. The most number of large packets are distributed in eMBB. While, the packet size of URLLC and mMTC are relatively smaller.

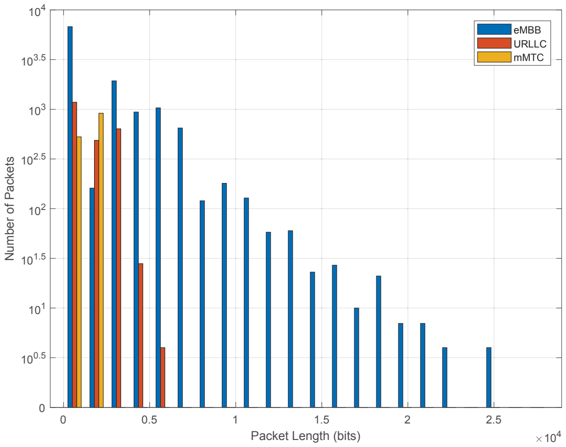


Fig. 4: Packet length of three service types.

The service type is selected based on the output of our NN. Then, the corresponding 3C resources are allocated. In our experiment, the accuracy of NN achieves 99.8% with the average training time of 109625 cycles. Thus, along with the effective data preprocessing techniques, the service types of the requests can be classified accurately and effectively. The distribution of 3C resources in network nodes affects the resource allocation decision of Controller. The total number of CPU cycles for completing service tasks is used to describe the computing resources of network nodes [21]. The caching resources of one network node is defined as the total number of contents that can be cached on it. The bandwidth and power of the network node are defined as the communication resources accordingly [5]. As shown in Fig. 6, the requirements for computing and caching resources of different service types vary in our experiment. eMBB requires the most caching resources, on the other hand, URLLC requires the most computing resources. eMBB supports high data rate services that require a large amount of communication resources. Thus, in the resource allocation process, we configure the highest bandwidth and power for this service type. Moreover, in order to improve the data rate during transmission, some popular contents are usually cached in the MEC to facilitate the transmission of content with high data rate when the content is requested. Therefore, more static cache resources are configured for eMBB to cache contents [20]. Moreover, URLLC and mMTC are allocated less static caching resource. URLLC supports intermittent transmission services. Meanwhile, URLLC also demands low latency and high reliability. Combined with its features, we configure higher bandwidth and higher power for it [15]. At the same time, we configure biggest computing resource for it to lower down latency of it. The main features of the mMTC service type are the transmission of small packets and lower latency

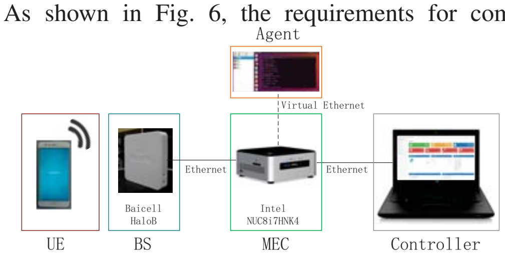


Fig. 5: Testbed of system.

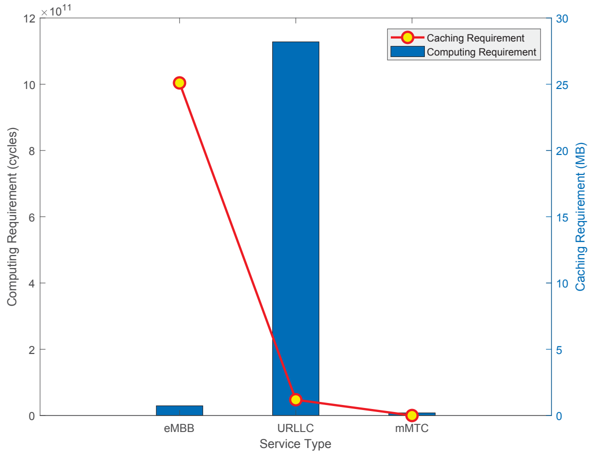


Fig. 6: Requirement of computing and caching resources.

sensitivity. Considering all above, this service type gets the lowest bandwidth and power configuration.

B. Implementation Result

During a service request of a UE, the inputs of our NN shown in Section III are obtained from our testbed shown in Fig. 5.

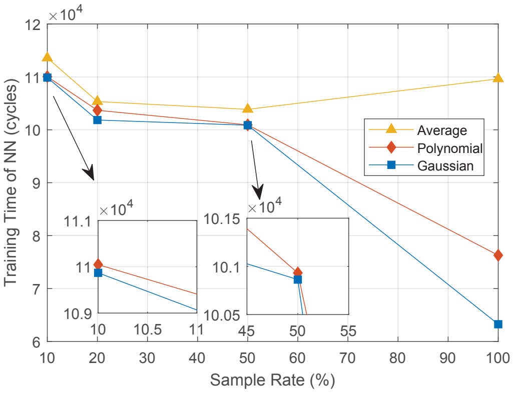


Fig. 7: Neural network model for resource allocation scheme.

Because of the intricate of the real environment, the original data usually contains noise. Thus it is necessary to filter noise which will improve the performance of the resource allocation scheme. We performed sampling and function fitting on the original data to filter the noise in the environment to some extent. Fig. 7 demonstrates the change of the training time of the neural network with the increase of the scale of the training sample data. The x-axis represents the sampling rate of packets during a service. The y-axis represents the training time of our NN. Specifically, in the pre-process stage of Algorithm 1, three methods, which are Gaussian fitting method, Polynomial fitting method, and Average method, are adopted for obtaining the statistics of the related inputs in the NN shown in Section III. Based on the experiment results shown in Fig. 7, we have the following observations. First, it is obvious that with the increase of the scale of the training sample data the training time of our NN of Gaussian fitting method keeps the trend of monotonous decreasing. Second, with the increase of sampling rate, the Average method is not monotonically decreasing, because there is noise in the real test environment, sampling can reduce the impact of noise to some extent. The polynomial fitting method is worse than Gaussian fitting method. Because, compared with Gaussian fitting method, polynomial fitting method is quite different from the data characteristics. Finally, we also can observe that the performance of the Average method is the worst under the same sample rate.

V. Conclusion

This work proposed a MEC and SDN technology based NS architecture in 5G networks to support the network optimization. Considering multidimensional resources, a resource allocation scheme based on the NN technology was proposed. It can be embedded into the NS architecture to allocate differentiated resources for diversified network services through NS technology By using the NS architecture, the network can obtain the resource allocation strategies quickly and effectively according to the service demands and the network status. The experiment on the test platform demonstrated the feasibility and accuracy of proposed architecture. The future work may focus on investigating various pre-process method for sampling data to improve the NN training accuracy.

References

[1] N. Alliance, "5G white paper", *Next generation mobile networks, white paper*, pp. 1-125, Feb. 2015.

[2] M. ITU-R, "Minimum requirements related to technical performance for IMT-2020 radio interface(s)," Nov. 2017.

[3] Cisco visual networking index: Global mobile data traffic forecast, *Cisco Public Information*, Feb. 18, 2019.

[4] N. Feamster, J. Rexford, and E. Zegura, "The road to SDN: An intellectual history of programmable networks," *ACM SIGCOMM Computer Communication Review,* vol. 44, no. 2, pp. 87-98, Apr. 2014.

[5] P. Popovski, K. F. Trillingsgaard, O. Simeone, and G. Durisi, "5G wireless network slicing for eMBB, URLLC, and mMTC: A communication-theoretic view," *IEEE Access,* vol. 6, pp. 55765-55779, Sep. 2018.

[6] I. AlQerm and B. Shihada, "A cooperative online learning scheme for resource allocation in 5G systems," in *Proc. IEEE International Conference on Communications (ICC)*, Kuala Lumpur, May 22-27, 2016.

[7] Z. H. Hussien and Y. Sadi, "Flexible radio resource allocation for machine type communications in 5G cellular networks," in *Proc. Signal Processing and Communications Applications Conference (SIU)*, Izmir, May 2-1, 2018.

[8] M. Dighriri, A. Saeed Dayem Alfoudi, G. Myoung Lee, T. Baker, and R. Pereira, "Resource allocation scheme in 5G network slices," in *Proc. International Conference on Advanced Information Networking and Applications Workshops (WAINA)*, Krakow, May 16-18, 2018.

[9] M. R. Raza, A. Rostami, L. Wosinska, and P. Monti, "A slice admission policy based on big data analytics for multi-tenant 5G networks," *Journal of Lightwave Technology*, vol. 37, no. 7, pp. 1690-1697, Apr. 2019.

[10] H. Halabian, "Distributed resource allocation optimization in 5G virtualized networks," *IEEE Journal on Selected Areas in Communications,* vol. 37, no. 3, pp. 627-642, Mar. 2019.

[11] R. Lippmann, "An introduction to computing with neural nets," *IEEE ASSP Magazine,* vol. 4, no. 2, pp. 4-22, Apr. 1987.

[12] M. ETSI, "Mobile edge computing (MEC); framework and reference architecture", *ETSI GS MEC 003*, V1.1.1, 2016.

[13] T. Mahmoodi, "5G and software-defined networking (SDN)," in *Proc. 5G Radio Technology Seminar: Exploring Technical Challenges in the Emerging 5G Ecosystem*, London, 2015.

[14] W. Cui, K. Shen, and W. Yu, "Spatial deep learning for wireless scheduling," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, Abu Dhabi, United Arab Emirates, Dec. 9-13, 2018.

[15] Ian Goodfellow, Yoshua Bengio, and Aaron Courville, *Deep Learning*, America, MIT Press, 2016.

[16] Christopher M. Bishop, *Neural Network for Pattern Recognition*, Cambridge, UK, Oxford University Press, 1995.

[17] M. Tang, L. Gao, and J. Huang, "Enabling edge cooperation in tactile internet via 3C resource sharing," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 11, pp. 2444-2454, Nov. 2018.

[18] E. K. Markakis, K. Karras, A. Sideris, G. Alexiou, and E. Pallis, "Computing, caching, and communication at the edge: The cornerstone for building a versatile 5G ecosystem," *IEEE Communications Magazine*, vol. 55, no. 11, pp. 152-157, Nov. 2017.

[19] V. P. Kafle, Y. Fukushima, P. Martinez-Julia, and T. Miyazawa, "Consideration on automation of 5G network slicing with machine learning," *2018 ITU Kaleidoscope: Machine Learning for a 5G Future (ITU K)*, Santa Fe, 2018, pp. 1-8.

[20] T. D. Tran and L. B. Le, "Joint resource allocation and content caching in virtualized content-centric wireless networks," *IEEE Access*, vol. 6, pp. 11329-11341, 2018.

[21] H. Mei, K. Wang, and K. Yang, "Multi-layer cloud-RAN with cooperative resource allocations for low-latency computing and communication services," *IEEE Access*, vol. 5, pp. 19023-19032, 2017.

[22] M. Tang, L. Gao, and J. Huang, "A general framework for crowdsourcing mobile communication, computation, and caching," in *Proc IEEE Global Communications Conference, Singapore*, 2017.

[23] Y. He, C. Liang, F. R. Yu, and V. C. M. Leung, "Integrated computing, caching, and communication for trust-based social networks: A big data DRL approach," in *Proc IEEE Global Communications Conference (GLOBECOM), Abu Dhabi, United Arab Emirates*, 2018.