2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring) | 978-1-7281-8964-2/20/$31.00 ©2021 IEEE | DOI: 10.1109/VTC2021-Spring51267.2021.9449052

# Optimize efficiency of Orchestration in Virtualized Radio Access Network Functions

lei xiao
*Chongqing College of Electronic Engineering*
Chongqing, China
21089945@qq.com

Sai Zou
*Chongqing College of Electronic Engineering*
Chongqing, China
dr-zousai@foxmail.com

***Abstract*—With the development of informatization, network businesses are expanding and business functions are becoming more powerful; thus, network infrastructure has begun to face new challenges in serving businesses. Research regarding the dynamic deployment of network slices according to business requirements is urgently needed. To develop a realistic and complete view of service providers' decision-making process, adaptation costs must be considered. Function orchestration of virtualized radio access networks is a critical way for network slices to offer web apps customized service. In this paper, we develop a mathematically optimized model for function orchestration and propose an orchestration strategy using a form of particle swarm optimization (VNFPSO), in virtualized radio access network. Bearing in mind the openness of radio access network frameworks, discreteness of network functions, and proliferation of network traffic, we have enhanced the state-of-the-art PSO algorithm in terms of inertia weight, particle mutation, and factor learning to improve the speed of obtaining a global approximate optimal solution. Simulated experimental results show that this strategy could decrease rejection rates in virtualized radio access networks and improve the utility of network system resources.**

***Index Terms*—Virtualization; Network functions orchestration; PSO.**

## I. Introduction

With the development of network function virtualization (NFV) and software-defined networks (SDNs), NFV orchestration (NFVO) has come to be regarded as the core of future networks; this task is crucial to helping operators flexibly manage network systems and maximize the advantages of new technologies[1]. NFVO can standardize virtual networking functions to increase the interoperability of SDN elements. NFVO can also orchestrate resources, network services, and other functions, rendering it a central component of an NFV-based solution. NFVO binds different functions to create an end-to-end, resource-coordinated service in an otherwise dispersed NFV environment (i.e., virtual service) [1]. It is therefore important to ensure that adequate computer, storage, and network resources are available to provide network services. NFVO can coordinate, authorize, release, and engage resources independent of a specific virtual infrastructure manager. It also provides governance around NFV instances of resource sharing in the NFV infrastructure. The Management and Organization Working Group of the European Telecommunications Standards Institute (ETSI) has defined the NFV orchestrator [2].

Current NFVO methods (i.e., orchestrating node functions along with the SFC) mainly orchestrate the corresponding infrastructure for specific web apps at specific times to maximize the global value of infrastructure [3]. NFVO is an emerging paradigm in which loosely coupled VNFs are orchestrated, located, and invoked on virtualized services as base stations in radio access networks. We have proposed a two-stage mapping framework for virtualized networks [4]. This framework first applies logical mapping to web apps and virtualized functions that are customized for web apps [5]; then, it applies physical mapping to the virtualized function set and infrastructure to maximize infrastructure use [6-9]. In [5], matching virtualized services for web apps are presumably designed by network management experts. In management platforms for virtualized radio access, virtualized network functions should be orchestrated automatically.

In recent years, telecommunication service providers have experienced a consistent decline in revenue [10-11]. NFV promises to decrease the costs of deploying and operating large networks by migrating network functions from dedicated hardware appliances to software instances running on general-purpose virtualized networking and computing infrastructures. It is generally believed that NFVO could lower equipment purchases and maintenance expenses without compromising flexibility and openness [12]. Therefore, in this paper, we prioritize economic benefits (without considering the sequence of virtualized network functions) when researching virtualized network resource orchestration. In actual network maintenance, the topologies of network change and resources for network adjustment require results from the business layers. Furthermore, business control and analysis cannot be accomplished without a thorough understanding of the resource

This work is supported by the National Natural Science Foundation of China (U2033212), the National Key Research and Development Project of China (2020YFB1711000), the National Scientific Research Foundation of Chongqing (cstc2019jcyj-msxmX0509), and the Social Scientific Research Foundation of China (19VSZ084), and the National Scientific Research Foundation of Chongqing Municipal Education Commission (KJQN201803110, KJZD-K201903101), and the Social Scientific Research Foundation of Chongqing Municipal Education Commission (20SKGH313), and the National Scientific Research Foundation of Hunan Province Education Commission (18B367), and the Youth Innovation talent Program of Guangxi(AD19245156), and the Teaching Reform Project of Guangxi Normal University(2019XJGZ08).

layer and relevant complexities. Two aims should thus be achieved to realize virtualized network function orchestration: customized functions of web apps and optimal economic benefits for infrastructure. Specific conditions should be satisfied to orchestrate virtualized network functions. The infrastructure will be managed and controlled by the virtualization layer, which can abstract the infrastructure into virtualized network functions without platform restrictions while representing the running states of physical resources. Based on the above analysis, we have conducted research to orchestrate the most economical virtualized service to be matched with web apps. Our main contributions are as follows:

- A mathematical model is proposed for virtualized networks to centrally choose relevant network components to orchestrate a highly economical new virtualized service.
- A novel modified particle swarm optimization algorithm (VNFPSO) is presented to solve our proposed model with a resource allocation problem.
- A simulated experiment is designed, through which the qualities of the VNFPSO algorithm in virtualized function orchestration are validated.

The remainder of this paper is organized as follows. In Section 2, we describe the system model and formulate the mathematical model. In Section 3, we describe our scheduling algorithms. The results of our experiment are presented in Section 4. Section 5 concludes our work.

## II. System Model and Model Formalization

### A. System model

Our cloud scenario is similar to the PaaS model, in which users can submit complex requests consisting of off-the-shelf services [13]. Off-the-shelf services have a highly convenient interface (e.g., API) to be called. Each service is associated with a price, which is assigned by its creator. When a user submits a compute request (or task) that calls other services, he/she must pay for usage of these services; the payment is determined by how many resources will be consumed. A specific virtualized service is orchestrated for every type of request. Moreover, every virtualized service shares infrastructure resources through time slots, as illustrated in Fig 1. We mainly focus on orchestration from requests to virtualized services; mapping research from virtualized services to infrastructure is described in [11-14].

We considered the environment for the edge cloud scenario. The whole system is divided into RF and signal processing parts. RF only retains part of power domain and frequency domain, and is virtualized and dynamically controlled. The signal processing part is transferred to the server and controlled by the virtual machine.The network functions of each virtual service are composed of power domain, frequency domain and server resources. The resources required for each virtual service can be supplied by either one physical node or multiple physical nodes.Each physical node can provide resources to either one virtual service or multiple virtual services.

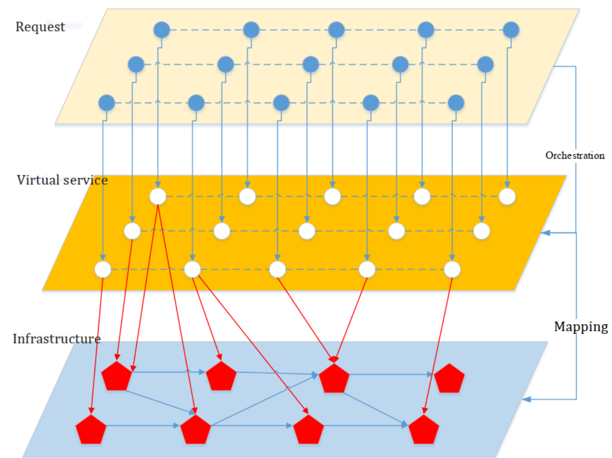


Fig. 1. Virtualized service principles

### B. Model Formulations

*1) Efficiency formulation of virtualized requests:* We define the virtualized request of web app $i$ as

$$R_i = (F_i, QoS_i) \tag{1}$$

, where $F_i$ is a virtualized function set requested by web app $i$, expressed as follows:

$$\begin{aligned} &F_i = (f_1, f_j, f_n) \\ &where\ f_j = \{id, name, description, note\} \end{aligned} \tag{2}$$

. $QoS_i$ is a corresponding feature of virtualized function $f_j$ requested by web app $i$, expressed as

$$\begin{aligned} &QoS_i = (a_1, a_k, a_m) \\ &where\ a_k = \{it^k, b^k, p^k\} \end{aligned} \tag{3}$$

. $f_j$ refers to the $j^{th}$ network function. $a_k$ denotes the features elicited by virtual function $f_j$. $it$ represents server resources of information technology, $b$ represents bandwidth resources, and $p$ represents power transceiver resources. $n$ denotes the number of functions in the resource pool, and $m$ denotes the number of features in the resource pool.

*2) Physical network benefits:* We can determine the cost of a unit spectrum to be $c_s$. The number $n_s^{i,j,k}$ of bandwidth resources on feature $a_k$ of virtualized function $f_j$ of web app $i$ can be written as

$$n_s^{i,j,k} = \left\lceil \frac{b^k}{B} \right\rceil \tag{4}$$

, where $B$ is the size of the unit bit-rate. The cost of spectrum utilization on radio access link $\eta_s$ is

$$\eta_s = n_s^{i,j,k} \times \exp\left[\frac{B}{\log_2(1+\gamma_i)}\right] \times c_s \tag{5}$$

, where $\gamma_i$ denotes the signal-to-noise ratio. We determine the cost of a unit power transceiver to be $c_p$. The cost of power transceiver utilization on radio access link $\eta_p$ is

$$\eta_p = p^k \times c_p \tag{6}$$

, We determine the cost of a unit resource of server $it$ as $c_{it}$. The cost of server resource $it$ consumption $\eta_{it}$ is

$$\eta_{it} = it^k \times c_{it} \tag{7}$$

, The deployment benefits of VNF comprise the joint utility function of spectrum utilization together with the power transceiver and server resources. The total cost of VNF deployment $\eta_d$ is

$$\eta_d = \sigma_b \times \eta_s + \sigma_p \times \eta_p + \sigma_{it} \times \eta_{it} \tag{8}$$

, where $\sigma_b$, $\sigma_p$, and $\sigma_{it}$ are the respective weighting factors, which satisfy $0 \le \sigma_b, \sigma_p, \sigma_{it} \le 1$ and $\sigma_b + \sigma_p + \sigma_{it} = 1$.

### C. Establishment of mathematical models

The optimization objective is to minimize the SP's overall orchestration cost. The optimized model can be written as follows:

$$\begin{aligned}
&obj \quad \min \sum_{j=1}^{n} \sum_{k=1}^{m} \mu_{j,k} \\
&s.t. \quad \begin{aligned}
&R_{i,j,k}.it \le N \times x_{j',k'}.it \\
&R_{i,j,k}.p \le N \times x_{j',k'}.p \\
&R_{i,j,k}.b \le N \times x_{j',k'}.b \\
&\forall x_{i,j} = 1, \exists x_{i+y,j} = 1,\ if\ f_i \to f_{i+y} \\
&\forall x_{i,j} = 1, \exists x_{i+y,j} = 0,\ if\ f_i \nrightarrow f_{i+y} \\
&dj = \delta_s \times \eta_s + \delta_p \times \eta_p + \delta_{it} \times \eta_{it} + \eta_d \\
&\mu_{j,k} = \begin{cases} dj & N = 1 \\ N \times dj + cost_{j,k} & N > 1 \\ N \times dj & 0 < N < 1 \\ 0 & other \end{cases}
\end{aligned}
\end{aligned} \tag{9}$$

, where $\mu_{j,k}$ is a value function, specifically the cost that virtual functional module $f_j$ with feature $a_k$ should pay. $x_{j',k'}$ is the selected functional module. $R_{i,j,k}.it \le N \times x_{j',k'}.it$, $R_{i,j,k}.p \le N \times x_{j',k'}.p$, $R_{i,j,k}.b \le N \times x_{j',k'}.b$ indicating that $N$ virtual functional modules $f_j$ with feature $a_k$ are chosen. Their server resources, spectrum resources, and sending and receiving power are equal to or greater than the resources requested by web app $i$. $f_i \to f_{i+y}$ indicates that virtual functional module $f_j$ and virtual functional module $f_{i+y}$ are mutually dependent; if $f_j$ exists, then there must also be $f_{i+y}$. $f_i \nrightarrow f_{i+y}$ indicates that virtual functional module $f_j$ and virtual functional module $f_{i+y}$ are mutually exclusive; if $f_j$ exists, then there must be no $f_{i+y}$. $cost_{j,k}$ represents the required expenses when $N$ virtual functional modules $f_j$ with feature $a_j$ are being used in parallel. $Mcost_{j,k}$ represents the required expenses when$N$ virtual functional modules $f_j$ with feature $a_j$ are sharing a resource. $\delta_s$, $\delta_p$, and $\delta_{it}$ are combined as the coefficient of functional module $x_{j',k'}$.

## III. Problem solving

The orchestration problem of the most economical service is an issue of combination optimization and classified as a maximum clique problem (MCP), which is NP hard. Two types of algorithms are available to solve MCP: certain algorithms and heuristic algorithms. Certain algorithms include the backtracking algorithm and 'red and black'. Certain algorithms can search any or all solutions to a problem systematically while remaining both systematic and skipping. However, the orchestration problem of virtual service is an NP complete problem, wherein certain algorithms are inherently time-consuming. Heuristic algorithms can obtain the approximate optimal solution more quickly. PSO is comparatively simple with a high convergence speed and offers an efficient method for solving MCP; however, the PSO algorithm suffers from pitfalls such as premature convergence, dimension problems, and local extreme values. Virtualization radio access networks are characterized by openness of the radio access network framework, discrete network functions, and exponential network traffic. In this paper, we strive to enhance the state-of-the-art PSO algorithm and propose the VNFPSO algorithm to solve the orchestration problem to provide the most economical service.

Assume that in the total dimension number $D = m \times n$ of orchestration problem, select any $L$ particles to form a group in which $i^{th}$ $(i < L)$ particle in $k$ generation can be described by two indexes; The value of virtual functional feature can be regarded as PSO location, represented as the $D$ dimensional vector of $Z_i^k = \left(z_{i,1}^k, z_{i,2}^k, \cdots, z_{i,D}^k\right)$; the change speed of virtualized function values can be seen as flying speed of PSO, repented as the $D$ dimensional vector of $V_i^k = \left(v_{i,1}^k, v_{i,2}^k, \cdots, v_{i,D}^k\right)$. The best location in individual history is $Ps_i^k = \left(ps_{i,1}^k, ps_{i,2}^k, \cdots, ps_{i,D}^k\right)$ when $i^{th}$ particle is searched to $k$ generation. The historical optimal location of the whole particle group is $Ps_g^k = \left(ps_{g,1}^k, ps_{g,2}^k, \cdots, ps_{g,D}^k\right)$ when is searched to $k$ generation. Thus the iteration formula of speed and location for $i^{th}$ particle at $j^{th}$ dimension is as follows:

$$\begin{cases} v_{i,j}^{k+1} = \begin{aligned}[t] &\omega \times v_{i,j}^k + c_1 \times r_1 \times \left(ps_{i,j}^k - z_{i,j}^k\right) \\ &+ c_2 \times r_2 \times \left(ps_{g,j}^k - z_{i,j}^k\right) \end{aligned} \\ z_{i,j}^{k+1} = z_{i,j}^k + v_{i,j}^{k+1} \end{cases} \tag{10}$$

, where $\omega$ is the inertia weight, representing the influence of prior speed on subsequent movement. $c_1$ and $c_2$ are learning factors; $r_1$ and $r_2$ are random numbers within [0,1].

### A. Inertia weight design in various steps

As an important parameter in the PSO algorithm, inertia weight $\omega$ plays a major role in balancing the convergence speed and search ability of optimal solutions. A larger $\omega$ was deemed favorable for convergence speed but could miss the optimal solution; this pattern suggests that a smaller $\omega$ may improve the local mining capacity of the algorithm but prolong convergence time. In accordance with this process, we can dynamically adjust inertia weight $\omega$ of the traditional PSO algorithm. When a particle location approaches the historical optimal location of individuals or the whole particle group, the convergence speed and flying speed should be enhanced. To guarantee a favorable local search capacity and convergence speed, the formula for inertia weight $\omega$ is

$$\omega^{k+1}=\begin{cases}\omega^k\times\left(1+\dfrac{1}{\sqrt{\sum_{m=1}^{2}\frac{\sum_{i=0}^{k}(c_{m,i})^2+\sum_{i=0}^{k}(r_{m,i})^2}{k^2}}}\right), \\ |ps_{i,j}^k-z_{i,j}^k|>tv_1\ or\ |ps_{g,j}^k-z_{g,j}^k|>tv_2 \\ \omega^k\times\left(1-\dfrac{1}{\sqrt{\sum_{m=1}^{2}\frac{\sum_{i=0}^{k}(c_{m,i})^2+\sum_{i=0}^{k}(r_{m,i})^2}{k^2}}}\right), \\ |ps_{i,j}^k-z_{i,j}^k|\le tv_1\ or\ |ps_{g,j}^k-z_{g,j}^k|\le tv_2\end{cases} \tag{11}$$

where $tv_1$ and $tv_2$ are threshold values, $c_{1,i}$ and $c_{2,i}$ are the $i^{th}$ learning factors, and $r_{1,i}$ and $r_{2,i}$ are the $i^{th}$ random functions.

### B. Mutation particle design

During the optimization process of the PSO algorithm, contradictions emerge between population diversity and algorithm convergence speed. The traditional PSO algorithm can be improved to enhance the local search capacity while maintaining population diversity to prevent premature convergence at higher speeds. The actual improved strategies are as follows:

Step 1: Select mutation particles If $ps_g^k-ps_i^k\ge tp$ , then $ps_i$ is a mutation particle and $tp$ is the threshold value.

Step 2: Smooth the moving distance of mutation particles

$$\begin{cases}Bd(1)=\dfrac{\sum_{i=0}^{L}|ps_i^1-ps_g^1|}{L} & k=1\\ Bd(k)=\dfrac{2\times\frac{\sum_{i=0}^{L}|ps_i^k-ps_g^k|}{L}+(k-1)\times Bd(k-1)}{k+1} & k\ge 2\end{cases} \tag{12}$$

. Here, $Bd(k)$ is the smooth-move distance of $k^{th}$ generation mutation particles.

Step 3: Particle mutation formula

$$ps_i^k=ps_i^k+Bd(k) \tag{13}$$

### C. Non-linear dynamic learning factor design

The formula to update the speed of the PSO algorithm consists of two parts. Under the influence of new speed $v_i$, a particle will gradually fly to a random weighted location between optimal global location $Ps_g$ and optimal partial location $ps_i$. The learning factors respectively represent the degree to which the partial location relies on its experience and the population's experience. The standard PSO algorithm generally sets the value of these two learning factors as a constant and considers the degree to be the same; however, every particle plays a different role at different stages. If we could guarantee that each particle could obtain a high degree of confidence initially with strong spatial development abilities, then revolutionized particles would have greater confidence in population decisions to aid in the particle population's convergence into either a global optimal or second optimal solution. This paper adopts non-linear dynamic learning factors based on triangle functions to manage the local development and global convergence abilities of particles. The value modifications of $c_1$ and $c_2$ are as follows:

$$c_1^{k+1}=c_1^k+\cos^2(c_1^k\times\frac{k}{T})+\beta\times\frac{\alpha}{\vartheta} \tag{14}$$

,

$$c_2^{k+1}=c_2^k+\sin^2(c_2^k\times\frac{k}{T})+\beta\times\frac{\alpha}{\vartheta} \tag{15}$$

, where $T$ is the total revolutionized algebra, and $\alpha$ is the evenly distributed random number within $(0,1)$. $\vartheta\in[2,10]$ is used to control the extent of shaking in random number $\alpha$. $\beta$ is either 0 or 1 and decides whether a random disturbance quantity $\alpha$ is added to $c_1$ or $c_2$ during the dynamic change process. $r_1=r_2=\frac{\alpha}{\vartheta}$. On the condition that $c_1+c_2$ is essentially stable, learning factor $c_1$ declines dynamically and $c_2$ increases dynamically.

### D. Experiment environment

During the entire simulation, we mainly focus on orchestrating nine virtualized network functions, each of which has four features that consume certain amounts of three resources: server resources, frequency resources, and power resources. Assuming that the minimum consumed resources for each feature are fixed (i.e., initially generated at random from 0 to 15), the total amount of server resources, frequency resources, and power resources is also fixed at 5000 each. The virtualized network function type and the features of function types differ; one or more features are chosen at random from each function, and the number is set at random as well. The aim is to determine the lowest-cost strategy to identify the number of features per function.

- Rejection rate of virtualized requests

The rejection rate of virtualized requests is represented by the ratio of rejected requests to requests. A request can be rejected in two scenarios: when the number of requested resources exceeds that in the service pool; or when the number of certain requested sources for a certain function exceeds the amount of resources available. Without considering business access and output, an experimental simulation was designed under the following conditions: low probability of web apps in total virtualized requests, 50% for web apps in total virtualized requests, and random web app generation; results appear in Fig 2.

"Traditional" refers to the traditional solution wherein the resources required for each function are fixed. When the resource number of virtualized requests required for each function is lower than the number of resources in the traditional solution, the service will be offered. "No pricing VNFPSO" involves VNFPSO, but the price for each resource is fixed. "Pricing VNFPSO" adopts VNFPSO, but the price of each resource is related to supply and demand for resources, as shown in Formulas (10) to (12). As shown in Fig 2, when the emergence rate of new apps was 50%, the fluctuation rate for the 0 to 20th web app was large in the traditional algorithm; the rejection rate was 0 for the 'no pricing VNFPSO' and 'pricing VNFPSO' algorithms. Beginning with the 21st web

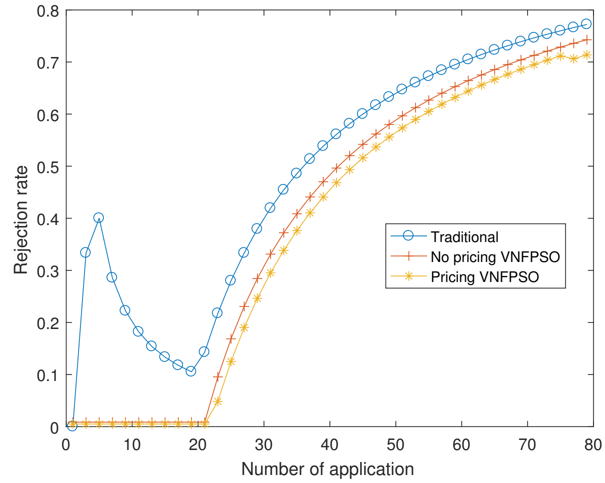


Fig. 2. Rejection rate of virtualized requests when emergence rate of new web apps is 50%

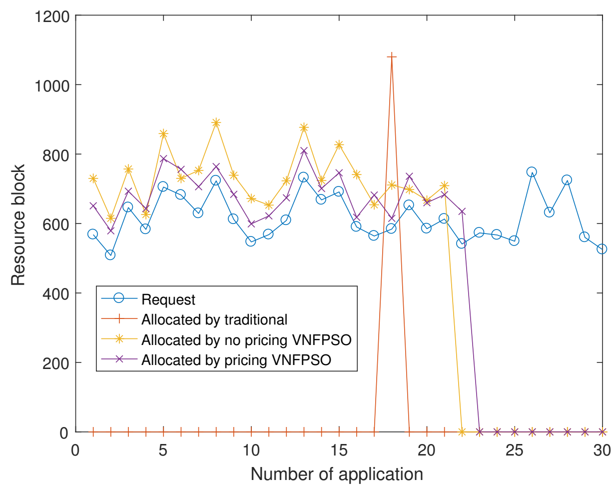


Fig. 3. Requested number and allocated number of resources

app, the rejection rate of these three algorithms was nearly the same.

- Resource number of virtualized requests

In Fig 3, "Request" denotes the total number of resources requested by each virtualized request; "Allocated by traditional" represents the actual allocation scenario of every virtualized overall resource in the traditional algorithm; and "Allocated by no pricing VNFPSO" is the actual allocation circumstance of every virtualized overall resource in 'pricing VNFPSO'. The traditional algorithm offered only one service for the first 30 requests, and the actual allocated resource amount exceeded the virtualized request resources. 'Pricing VNFPSO' offered 23 services at most for the first 30 virtualized requests, and the actual allocated resource amount was close to the virtualized request resources. 'No pricing VNFPSO' was not as effective as 'pricing VNFPSO'.

## IV. Conclusion

Network slicing is the basis of solving the problem of diversified demands in future networks. VNFO is a core technology in network slicing. We have studied virtualized radio access resource orchestration. We established a mathematical orchestration model for virtualized network functions based on economic benefits. This work solves the orchestration problem of inner resources for each virtualized network. Subsequent research will involve automatic orchestration of virtualized services, namely by selecting a virtualized network function set, ordering and processing each virtualized network function, and examining connection problems between each function.

## References


[1] SDxCentral, What Is An NFV Orchestrator (NFVO)? Definition, https://www.sdxcentral.com/nfv/definitions/nfv-orchestrator-nfvo-definition/, 2019

[2] ETSI, Network Functions Virtualisation (NFV): Management and Orchestration, http://www.etsi.org/deliver/etsi_gs/NFV-MAN/001_099/001/01.01.01_60/gs_NFV- MAN001v010101p.pdf. 2017

[3] Riggio, Roberto, et al. Scheduling Wireless Virtual Networks Functions. IEEE Transactions on Network & Service Management , 13.2(2016):240-252.

[4] Zou Sai, et al. RSCMF : A Resource Mapping Framework for Scalable Collaboration in Radio Access Networks with SDR and Virtualization, Journal Of Internet Technology, Doi: 10.3966/160792642019032002013, 2019

[5] Zou, Sai, et al. An Identification Decision Tree Learning Model for Self-Management in Virtual Radio Access Network: IDTLM. IEEE Access, 6(2018):504-518.

[6] Zou, Sai, et al. Resource Multi-Objective Mapping Algorithm Based on Virtualized Network Functions: RMMA. Applied Soft Computing, (2018):S1568494618300346.

[7] Zou Sai, et al. The Resource Mapping Algorithm of Wireless Virtualized Network for Saving Energy in Super Dense Small Cells. Mobile Information Systems. Http://Dx.Doi.Org/10.1155/2015/958431 ,2015.12 .

[8] Zou, Sai, et al. Resource Allocation Mechanism Based on Two-Step Mapping for Saving Energy in Wireless Network Virtualization. IEEE International Conference on Anti-Counterfeiting 2015.

[9] Zou, Sai, et al. A Novel Algorithm of Virtual Resource Allocation in Heterogeneous Radio Access Networks. International Conference on Computer Science & Education IEEE, 2016.

[10] Checko, Aleksandra , et al. Cloud RAN for Mobile Networks: A Technology Overview. IEEE Communications Surveys & Tutorials, 17.1(2015):405-426.

[11] Mijumbi, Rashid, et al. Management and Orchestration Challenges in Network Function Virtualization. IEEE Communications Magazine, 54.1(2016):98-105.

[12] Mijumbi, Rashid, et al. Network Function Virtualization: State-Of-The-Art and Research Challenges. IEEE Communications Surveys & Tutorials18.1, (2017):236-262.

[13] Di, Sheng, et al. Optimization of Composite Cloud Service Processing With Virtual Machines. IEEE Transactions on Computers, 64.6(2015):1755-1768.