RESEARCH ARTICLE OPEN ACCESS

Cost-efficient Resource Schedule Scheme Based on Packet Dropping Positions in Service Chain

Chengxiang Li *
*(Department of Computer science, Jinan University, Guangzhou 510632, China
Email: mirrorforesting@gmail.com)

--************************----------------------------------

Abstract:

Network Function virtualization(NFV) as an emerging technology, it enables dedicated hardware for network functions to transform to software called Virtual Network Function(VNF) with more convenience and elasticity. VNFs can be chained together in given order as service chain to provide a specific network service. When multiple VNFs are consolidated into single server, limited computing resource would be used bymore than onecompetitor, unreasonable resource scheduling scheme can lead to severe resource wastage. What's more, in the environment of service chain, packets be dropped at a certain position has been processed by upstream VNFs, so losing packets at different positions of service chain can cause different degree of resourcecost. This paper studies the resource scheduling scheme of service chain in light of this issue. Firstly, packets loss cost model is defined base on queue theory, and to minimize cost of losing packets, a greedy based cost-efficient algorithm which is sensitive to positions and arrival packet rates is proposed after analysis. Simulation shows that our proposed algorithm achieves optimization on reducing cost by up to 74.93%compared with other algorithmswhen seven VNFs in service chain. Evaluation in our paper demonstrates thatgreedy based cost-efficient algorithmoutperforms other competitorsobviously.

Keywords **—NFV, service chain, resource schedule.**

--************************----------------------------------

I. INTRODUCTION

Network Functions such as firewall, traffic monitor and load balancer that coupled with proprietary devices are widely used in today's network, which leads to difficulty in network management and service provision[1]. Network Function virtualization(NFV) is a new network architecture paradigm which makes it possible to change the way of deploying network function from dedicated hardware to software running on top of industry standard server by leveraging virtualization technology.Not only does NFV decrease capital expenditure of implementing network functions and operating, but also it increases flexibility and efficiency of deployment. And the network function which is realized in the form of software is referred to as Virtual Network Function(VNF).

At the same time, multiple VNFs can be chained together in a given order, working as service chain to provide a specific network service. It requires the data traffic to pass through a certain VNFs and packets are supposed to be processed by a sequence of VNFs before being forwarding to the destination[2].For the problem of service chain orchestration, one of main challenges is achieving the effective and reasonable resource scheduling under service demand[15], inappropriate scheduling scheme may cause considerable resource wastage.

Whenmultiple VNFs of service chains contend for the same computing resource on server, losing packets is a common phenomenon in network service. These packets in queues of VNF cannot be processed in time and overflowing occurs. In the environment of service chain, packets be droppedat a certain position have been processed and transmitted by upstream VNFs and the resources consumed on these packets would be wasted. It's worth noting that the cost of dropping packets at different positions of service chain are not same totally. For example, under the circumstance of service chains which contains three VNFs in server, the packets dropped at the first VNF are not processed by any VNFwhile the packets dropped at third VNF have been processed by upstream service chain and the relative resource has been consumed.Moreover, severe packets loss due to unreasonable resource scheduling would lead to inefficient network service performance, VNF will turn into a bottleneck of service chain if it drops too much packetsunder some kind of resource scheduling mechanism, and then the packet delay increases and the quality of network service cannot be guaranteed. In light of this, an efficient and specific resource scheduling policy is needed for service chain deployment in server.

Packets deliveredto a VNF in service chain have already been processed by all upstream VNFs with various resource cost. To minimize the cost caused by dropping packetsin service chain, in this paper, we propose a greedy-based resource scheduling algorithm to achieve more fairness and efficiency. Rather than equal share of CPU time slice, our proposed scheduling algorithmcan be sensitive to packet dropping positions, whichmakes the packet dropping cost as small as possible, at the same time, the service levelagreement of service chain such as latency and throughput can be satisfied.

The rest of this paper is as follows: Section II presents the related work and highlights the detail of other scheduling mechanisms and advantages of our motivation and concerns. Section III formulates the problem and Section IV describes the algorithm analysis and our proposed greedyheuristic schedulingmethod is discussed in this section. Performance evaluation results and comparison with selected state-of-art algorithms are reported in Section V.

II. RELATED WORK

As the promisingtechnology for network functions provisioning, NFV has attracted significant attention from both industry and academia. Multiple efforts have been focusing on resource scheduling in the environment of service chain.

Qu*et al.*[3] focus the effect of transmission and processing delay on resource scheduling, and transfers this problem into an ILP(Integer Linear Programming). Genetic algorithm based schedulingpolicy is proposed to solve the problem. Yao*et al.*[7] design a preemptive scheduling mechanism, and aims to minimize the system completion time when scheduling the CPU resources.Pham*et al.*[4] leverage the stablematchingtheory and treats the VNF and time slice as one-to-one matching game to achieve scheduling. Zhang*et al.*[5]study the circumstance of shared VNF instanceson the same node, then formulates this problem as an ILP model, the objective of this work is maximizing the number of service chain requests.

In addition, the authors of [8] emphasize the existing interactions of VNFs mapping, scheduling and traffic routing, a novel approach based on column generation has been proposed.Yoshida M*et al.*[9]consider the combination of possibly conflicting objectives with multifaceted constraints and proposea multi-objective resource scheduling algorithm to optimize the infrastructure resource.In [10], authorsdesign a heuristic method to coordinate the composition of VNF chains and its embedding into the substrate network in large scale.Yi B*et al.*[11] propose a weight-based VNF sharing scheduling approachin order to improve the resource utilization and reduce the resource fragmentation.

Although the solutions for resource scheduling in the circumstance of service chainsmentioned above are effective for theirobjectives, but actually these works ignorethat whether the packets of service chain requests are dropped or not, evenassume no packetsloss.Besides, the authors of [12] focus thepower consumption optimization, an efficient framework proposed allows to optimally assign and schedule flows to VNFs.Kulkarni S G*et al.*[2] design a dynamic resource scheduling platform of VNF deployment, computing resource can be allocated based on VNFs' loads, backpressure algorithm is used to control the packets arrival actively. But this work does not focus the packet loss in service chainand achieve resourceschedule with consideration on VNFs' positions. Li C *et al.*[13] discuss the cost of losing packets in service chain and LP-based approach is proposed to optimize the resource scheduling.This work aims to reduce the cost, but the definition of packets dropping cost is not practical, and the solution has high complexity in large deployment scale.

III. SYSTEM MODEL

This paper focuses on resource scheduling issue of service chain with given topology on server, in this section we present a formal model for this problem description.

A. VNF and Service Chain

We consider set of service chains which need to be deployed on server as $G = \{sc_1, sc_2, \ldots, sc_i\}, i \in [1, |G|]$, where sc_i represents the i^{th} service chain. $|G|$ is the number of service chains on server.Service chain sc_i here is modelled as a tuple, i.e. $[f_{i,1}, f_{i,2}, \ldots, f_{i,j}, b_i, w_i], j \in [1, |sc_i|]$,which contains the j^{th} VNF $f_{i,j}$ of service chain sc_i, the bandwidth requirement b_i of service chain sc_i and upper bound delay of whichcurrent service chain sc_i can be tolerant. $|sc_i|$ indicates the number of VNFs that sc_i has.Service chain requires a certain throughput of traffic processing, and packets are supposed totraverse the whole service chain within required time to guarantee the quality of network service. b_i and w_i ensures that service level agreement of service chains in resource scheduling.

In addition, for VNF $f_{i,j}$, packets arrive at it with a certain traffic rate, and different VNFs ranging from Monitor to Firewall need to process these arrival packets from a particular network service requestwith multiple purposes,and actions applied on the same packet by different VNFs are not same[14]. For example, flow monitor does not modify packet and performs log function generally. But if the same packet needs to be processed by IDS (Intrusion Detection System) for security check, the header or payload ofthis packet would be analysed according to rules configured.So, here let $\lambda_{i,j}$ be the packet arrival rate of $f_{i,j}$, and we use $\varphi_{i,j}$ todenote the processing capacityper unit of computing resource of $f_{i,j}$. Because VNFs have different types, under the same quota of CPU resource, the processing rate of VNFs with different types are not same.Besides, the maximalbuffer size of $f_{i,j}$ is denoted as $N_{i,j}$. It is worth to noting that the topology and composition of service chains on single server has been decided before schedule period. And the available computing resource of current server node is Φ.

B. Packets Loss Model

When multiple service chains are densely packed into single server, the computing resource could easilybe bottleneck[2].In this environment, losing packet is unavoidable during overload situation due to buffer overflow, it occursat a VNFwhenever the amount of traffic injected to that function exceeds its buffer capacity[16]. To quantify packet loss, we leverage queueing theory[17]. Assume that traffic flow arrives at VNFs is stable and continuous, according to[18]–[23], the traffic can be well modelled as Poisson stream, and the interval timeof consecutive arriving packet satisfies exponential distribution, the process on VNF of each packet is independent with each other. When packets of service chain request arrive at arbitrary VNF, these packets are served at fixed processing rate and follow FIFO (First In First Out) policy. In view of these, $M/M/1/N/\infty$ queueing model[17] can be applied to single VNF.

Based on $M/M/1/N/\infty$ model, assume the time slice ratio of computing resource allocated to $f_{i,j}$ is $r_{i,j}$, the quantity of lost packets $L_{i,j}$ within single schedule period ΔT can be formulated as follow:

For each VNF $f_{i,j}$, $\rho_{i,j}$ is the ratio of packets arrival rate $\lambda_{i,j}$ toprocessing rate $r_{i,j} \cdot \varphi_{i,j}$, where $\varphi_{i,j}$ is decided by VNFs' types, different VNFs' types are corresponding to different values. In terms of properties of queue model, the probability ofempty state $P_0^{i,j}$ and filled state $P_N^{i,j}$ can be defined as equation (3) and (2). $P_N^{i,j}$ is also the probability of rejecting rate of packets when the queue being filled. Combined withpackets arrival rate $\lambda_{i,j}$ andschedule period ΔT, the quantity of lost packets can be acquired as equation (1).

C. PacketDropping Cost

When a VNF becomes overloaded, packet dropping is inevitable. The dropped packets at a certain position of service chain have already been transmitted and processed by upstream VNFs, so the resource has been consumed by thesepackets would be wasted.Losing packets at different positions can lead to various degree of resource wastage of upstream service chain.Resource wastage caused by a single packet dropping at the first VNF of a service chain is totally different from the same dropping at the VNF which is located on the end of service chain[13].Generally, more cost would be caused by dropping at latter VNFs in service chain in that these packets have traversed more VNFs and links. In practice, the cost of dropping packets in service chain is also relevant to the upstream VNFs' types which can be reflected by the processing capacity per unit of computing resource $\varphi_{i,j}$. Under the same quota of computing resource $r_{i,j}$, thehigher the value of $\varphi_{i,j}$ is, the higherthe processing rate $f_{i,j}$ has, whilecomputingresourceneeds to be consumed on processing single packet can be less. So it can be found thatcomputing resource costof different VNFs forprocessing packetarerelated to itstypes, which are not same.

To formulatethe packet dropping cost of service chain, we focus the upstreamcost at a certain position firstly. Letaverage service latencyof packet in VNF be the indicator of resource consumption when processing packets, where service latency includes the queueing and processing delay.According to Little's formula in queue model[17],the average service latency which is also

$$L_{i,j} = \lambda_{i,j} \cdot P_N^{i,j} \cdot \Delta T \qquad (1)$$

$$P_N^{i,j} = \rho_{i,j}^{N_{i,j}} \cdot P_0^{i,j} \qquad (2)$$

$$P_0^{i,j} = \frac{1-\rho_{i,j}}{1-\rho_{i,j}^{N_{i,j}+1}} \qquad (3)$$

$$\rho_{i,j} = \frac{\lambda_{i,j}}{r_{i,j} \cdot \varphi_{i,j}} \qquad (4)$$

the packetcomputation cost of $f_{i,j}$ can be expressed as follow:

$$\delta_{i,j} = \frac{\frac{\rho_{i,j}}{1-\rho_{i,j}} - \frac{(N_{i,j}+1)\rho_{i,j}^{N_{i,j}+1}}{1-\rho_{i,j}^{N_{i,j}+1}}}{\lambda_{i,j}^{e}} \qquad (5)$$

In the equation (5), $\lambda_{i,j}^{e}$ is the effective packets arrival rate, i.e., actual acceptance rate when current VNF $f_{i,j}$ is not overload, it can be calculated as follow:

$$\lambda_{i,j}^{e} = \lambda_{i,j} \cdot (1 - P_N^{i,j}) \qquad (6)$$

For the reason that all VNFs are consolidated into a server, latency between adjacent VNF of service chaincaused byinterior communication is tiny, the cost of transmission between VNFs is not discussed in ourmodel.In this situation, the upstream resource cost $C_{i,j}$ of single packet dropped by $f_{i,j}$ can be defined as:

$$C_{i,j} = \begin{cases} C_{i,j-1} + \delta_{i,j-1}, & j > 2 \\ \delta_{i,1}, & j = 2 \\ 1, & j = 1 \end{cases} \qquad (7)$$

From equation (7), it is obvious that the upstream VNF of second VNF in service chain sc_i is $f_{i,1}$, so the upstream resource cost of single packet dropped by $f_{i,2}$ is its packet computation cost.When $j > 2$, this cost can be calculated by recursion method.

Then, the packet dropping cost of $f_{i,j}$ can be formulated as follow:

$$Cost_{i,j} = L_{i,j} \cdot C_{i,j} \qquad (8)$$

As shown in equation (8), when $j = 1$, the packet dropping cost of the first VNF is the quantity of lost packets $L_{i,1}$ itself. The cost of other positions' VNF is the product of the quantity of lost packets $L_{i,j}$ and the upstream resource cost $C_{i,j}$ of single packet dropped.

D. Shared VNF of Multiple Service Chain

VNF instance is often shared by multiple service chains when deploying. Under this circumstance, traffic from different network service requests arrive at the same VNF. We consider the packets flow split and merge based on former work[19]. The upstream resource cost can be divided by thearrival rates ofthe shared VNFfromdifferent service chains. Assume that $f_{i,j}$ and $f_{i',j'}$ is same VNF and sharedwith each other, the proportion of packet arrival rates from service chainsc_i and $sc_{i'}$ can be expressed respectivelyas follow:

$$\omega_1 = \frac{\lambda_{i,j}}{\lambda_{i,j} + \lambda_{i',j'}} \quad (9)$$

$$\omega_2 = \frac{\lambda_{i',j'}}{\lambda_{i,j} + \lambda_{i',j'}} \quad (10)$$

Because the packets from these two service chains can arrive at the same shared VNF's buffer.The packet dropping cost of this shared VNF is shown in equation (11).

$$Cost_{i,j} = L_{i,j} \cdot (\omega_1 \cdot C_{i,j} + \omega_2 \cdot C_{i',j'}) \quad (11)$$

For example, in Fig.1, the VNFs marked in yellow are shared by two service chains, which contains three and four VNFs respectively. Fig. 1 describes two scenarios: one is packets arrive at a shared VNFare from two different upstream service chains, another is two service chain requests has mergedandthe upstream VNFs contain shared VNF.

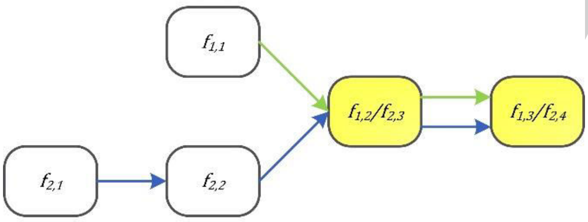

Fig. 1 Example of shared VNFs in multiple service chains

In this example, the packet dropping cost$Cost_{1,2}$ of $f_{1,2}/f_{2,3}$ can be calculated as:

$$L_{1,2} \cdot (\omega_1 \cdot \delta_{1,1} + \omega_2 \cdot (\delta_{2,1} + \delta_{2,2})) \quad (12)$$

$$\omega_1 = \frac{\lambda_{1,2}}{\lambda_{1,2} + \lambda_{2,3}} \quad (13)$$

$$\omega_2 = \frac{\lambda_{2,3}}{\lambda_{1,2} + \lambda_{2,3}} \quad (14)$$

And the another shared VNF$f_{1,3}/f_{2,4}$is expressed as:

$$L_{1,3} \cdot (\omega_1 \cdot \delta_{1,1} + \omega_2 \cdot \left(\delta_{2,1} + \delta_{2,2}\right) + \delta_{1,2}) \quad (15)$$

Equation (12)shows the definition of cost when the two different requests traffic arrive at $f_{1,2}$ directly. Equation (15) defines the cost when the number of shared VNFs is more than one.

As we can see in this example, if the $\lambda_{2,3}$ is empty, the number of service chains remaining in Fig. 1 is one which is marked in green. The packet dropping cost of $f_{1,3}$ can be transformed to:

$$L_{1,3} \cdot (\delta_{1,1} + \delta_{1,2}) \quad (16)$$

E. Packet Delay

Severe packet loss can lead to performance degradation of network services, especially the packet delay increases substantially, and the quality of services cannot be guaranteed.Besides, the upstream resource cost of the VNFs which are located at the back of the service chain arehigher than others generally.These VNFs tend to acquire more computing resource to reduce the total cost as much as possible. But the VNFs of same service chain which are at other positions also need sufficient resource to ensure the throughput and latency required.Therefore, each service chain has bandwidth requirement b_i and the upper bound of packet delay of which the sc_i can be tolerantw_i. Anefficient and reasonable resource scheme should take the quality of networkservicesinto consideration.

Here, additional constraintson packet delay are shown as follow:

$$r_{i,j} \cdot \varphi_{i,j} \geq b_i, \forall i \in [1, |G|], j \in [1, |sc_i|] \quad (16)$$

$$\sum\nolimits_{j \in [1,|sc_i|]} \delta_{i,j} \leq w_i, \forall i \in [1, |G|] \quad (17)$$

Equation (16) ensures that the resource allocated to $f_{i,j}$ should satisfy the traffic bandwidth requirement and equation(17) ensures thatthe packet delay of traversing the whole service chain is within the expected maximum value.

F. Schedule Objective

Assume that the total resource ratio is 100% which is corresponding tosingle scheduleperiodΔT. Equation (18) indicates the sum of resource time slices that have been allocated to$f_{i,j}$ is1, and this value can be modified on the basis of serverstatus. $\varepsilon_{i,j}$ is used to denote whether $f_{i,j}$ is shared by other service chainsor not,$\varepsilon_{i,j} = 0$ if$f_{i,j}$ uses the VNF of other service chains and has been counted.Otherwise, $\varepsilon_{i,j}$ is 1.

$$\sum_{i\in[1,|G|]}\sum_{j\in[1,|sc_i|]} \varepsilon_{i,j} r_{i,j} = 1 \qquad (18)$$

In our paper, the optimization objective can be specified as shown in equation (19).

$$\sum_{i\in[1,|G|]}\sum_{j\in[1,|sc_i|]} Cost_{i,j} \qquad (19)$$

$Cost_{i,j}$ should be minimized while the quality of network service need to be guaranteed.

IV. ALGORITHM DESIGN AND ANALYSIS

To minimize the total cost of packet dropping on service chain, the quantity of lost packets $L_{i,j}$ and the upstream resource cost $C_{i,j}$ should be as lower as possible.When the arrival rate of$f_{i,j}$is fixed, the more computing resource $f_{i,j}$ has, the higher processing rateit would own. The quantity of lost packets in ΔT can be reduced with sufficient processing capacity.The upstream resource cost$C_{i,j}$ is associated withservice latency$\delta_{i,j}$in VNF which includes queueing and processing delay.All VNFs in a service chain, except the last one, can be the role ofupstream VNFs. So the optimization on service latency in all VNFs is an important part of reducing upstream resource cost. This latency would decrease when processing rate increases under a certain arrival rate.

In addition, as we discussed above, the packets being dropped at a certain position of service chain have been processed by upstream VNFs, in general, thepacketsdropped by the VNF which is locatedat the back of service chainalways suffer more upstream resource wastage.So, these VNFs which are further back in service chains prefer to get more computing resource to reduce the quantity of lost packets and total cost as well.

The types of VNFs also have a perceptible effect on the total resource cost caused by packet dropping. VNFs with higher$\varphi_{i,j}$ are apt to ownmore processing ratethan others when the allocated time slices are same.Not only the quantity of lost packets but also the upstream resource cost on processing these packetswould be reducedif VNFs have more processing capacity.To reduce the total cost, more computing resource should be assigned to VNFs with lower$\varphi_{i,j}$.

To achieve resource scheduling withefficiency andfairness. Three factors which are packets arrival rate, VNFs' positions in service chain and VNFs' types should be taken into consideration. Based on the analysis above, a preference weight $\tau_{i,j}$can be defined as follow:

$$\tau_{i,j} = \frac{\lambda_{i,j} \cdot Pos_{i,j}}{\varphi_{i,j}} \qquad (20)$$

$Pos_{i,j}$ is the position weight of VNF $f_{i,j}$, it can be calculated by service chain topology,the position weight of $f_{i,j}$ is$\frac{j}{|sc_i|}$. The value of$\tau_{i,j}$ indicates that more computing resource prefers the VNF which has higher packet arrival rate, position weight and lower processing capacity perunit of computing resource.

In light of these, we propose a greedy-based solution on resource schedulingfor service chainson server.The algorithm is shown below.

Algorithm 1:Packet Dropping CostOptimization Algorithm (PDCO)

Input: $\Phi, G, b_i, \lambda_{i,j}, \varphi_{i,j}, N_{i,j}, \Delta T, i \in [1,|G|], j \in [1,|sc_i|]$
Output: $r_{i,j}$

```
Initialize:SetR ← ∅, s ← 1, M ← number of VNFs
For f_{i,j} ∈ G on server do
        R[i, j] ← b_i/φ_{i,j}
        Φ ← Φ − b_i/φ_{i,j}
        Pos_{i,j} ← j/|sc_i|
        τ_{i,j} ← λ_{i,j} · Pos_{i,j}/φ_{i,j}
End for
For f_{i,j} ∈ G sorted by τ_{i,j} in descending order do
If R[i, j] < λ_i/φ_{i,j} then
If Φ ≥ λ_i/φ_{i,j} − R[i, j] then
        Φ ← Φ − (λ_i/φ_{i,j} − R[i, j])
        R[i, j] ← λ_i/φ_{i,j}
Else
```

14: R[i, j] ← R[i, j] + Φ
15: Φ ← 0
16: **Endif**
17: **End if**
18: **If**Φ ≤ 0**then**
19: **break**
20: **End for**
21: **Return**R

For each VNF hasdeployed on server, the traffic bandwidth requirement of service chain to which it belongs should be satisfied. Firstly, as we can see from line 2 to 7 in algorithm 1, it describes the minimum resource that ensures bandwidth requirement must be allocated to all VNFs on server.After computing quantities of this resource for a VNF, available residual resource Φ should be changed to new value as line 4 shows. In the first loop in algorithm 1, the preference weights$\tau_{i,j}$ for $f_{i,j}$are also decided for the next period of resource scheduling.

From line 8 to 20, it shows thegreedy-based resource scheduling scheme for $f_{i,j}$ on single server.When the available resource is sufficient, algorithm 1 requires that VNF withhigher$\tau_{i,j}$ should be scheduled with more priorities(line 8).Efficient resource scheduling must be on the basis of $\lambda_{i,j}$ of all VNFs.Not only should the resource allocated satisfy the processing of arrival packets, but also resource wastage should be avoided, i.e., the resource for $f_{i,j}$ should not over$\lambda_{i,j}/\varphi_{i,j}$. Sowhen the resource have been allocated R[i, j] is less than$\lambda_{i,j}/\varphi_{i,j}$, available resource will be scheduled to $f_{i,j}$with priority according to $\tau_{i,j}$(line 9 to 17).If the current available resource Φ is sufficient for$f_{i,j}$ to own$\lambda_{i,j}/\varphi_{i,j}$ of resource(line 10), thenallocate resourcecatering for arrival packet rate$\lambda_{i,j}$ to reduce lost packet as much as possible(line 11 to 12). Otherwise, add Φto computing resource for$f_{i,j}$(line 14 to 15).

From algorithm 1, it can be easily found that the complexity of this algorithm is related to service chains scale. i.e., the number of VNFs deployed on server. In light of this, the complexity of algorithm 1 is known to be$O(M)$.

V. EVALUATION

To demonstrate the effectiveness of our proposed algorithm, numerical simulation is conducted to evaluate the performance of greedy-based resource scheduling scheme PDCO from several aspects. The competitors of this evaluation areMDC[13] from former work, greedy based on load of VNF (GL)[13]andRound-Robin method (RR)[24]. Besides, the number of deployed VNFs is set from 2 to 10, and the available residual resource is 100%,ΔT is 10ms. And the packet arrival rate$\lambda_{i,j}$ isgenerated randomlyfrom 600 to 800p/ms(packets per millisecond) while $N_{i,j}$ is 700. The bandwidth requirements of service chainb_i is set as 400 p/ms, and$\varphi_{i,j}$ which indicates VNFs' type is set from 20 to60p/ms per unit of computing resource.

Packet dropping cost as a crucial metric was evaluated firstly. The result is shown in Fig. 2. It demonstrates the performance of PDCO with regard to reducing packet dropping cost. In comparison, the proposed PDCO resource scheduling algorithm outperforms MDC by up to 74.93 percent when service chain contains seven VNFs with different types.Compared with other two scheduling algorithms, PDCO has significant advantage on cost optimization with any VNFs' scale.

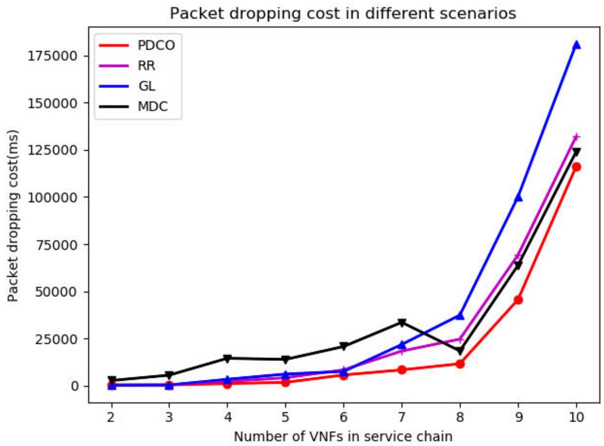

Fig. 2 Packet dropping cost in different scenarios

We also studied theperformance on packet loss rate of service chain and specific VNFs. As we can see from Fig. 3, the average packet loss rate is relatively high when the number of VNFs in service chain is ranging from 2 to 7 under MDC algorithm. And from 2 to 8 VNFs in service chains, PDCO performs best in all algorithms, with values of average packet loss rates not exceeding9.36 percent. It can be found that when the scale of VNFs turns to larger, the results on packet loss rates of these algorithms except GL become similar, because the quantity of available resource on single server is only 100% in our experiment, when the number of VNFs deployed on server increases to 10, the allocated resource for VNFs by these algorithms cannot satisfy all arrival traffic, while these resource scheduling schemesmust ensure the bandwidth requirement b_i of current service chain. Under this circumstance, part of schedulingalgorithms achieve similar results based on VNFs' types setting in this experiment.

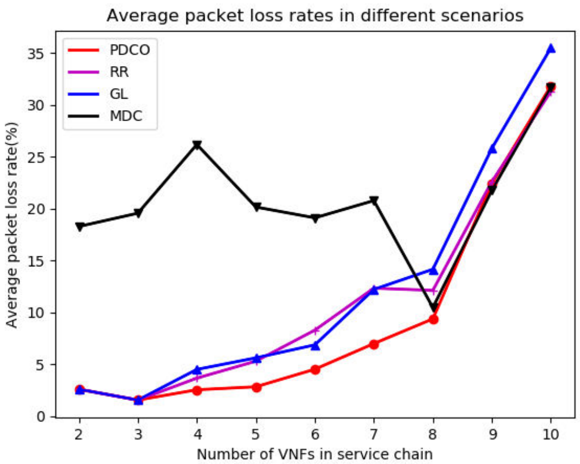

Fig. 3 Averagepacket loss rates in different scenarios

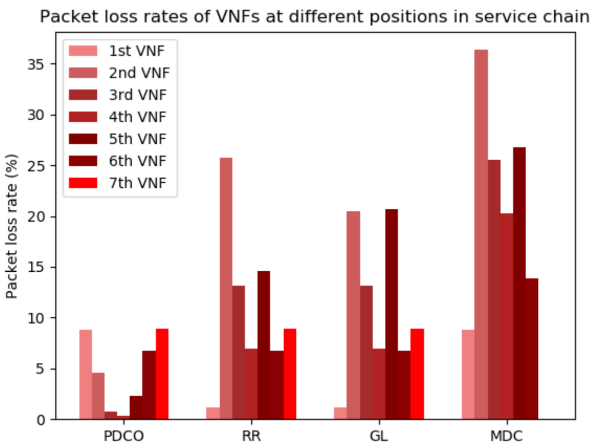

Fig. 4 Packet loss rates of VNFs at different positions in service chain

To study further on the metric of packet loss rate, we conducted an evaluation on specific VNFs. Fig. 4 shows the packet loss rates of VNFs at different positions in service chain which contains seven VNFs. We observed that PDCO outperforms all the other algorithms obviously.MDC aims to minimize the packet dropping cost by LP method, it typically allocates resource as much as possible to the lastVNF to reduce the lost packets which are related to more upstream resource cost. So, as we can see in Fig. 4, the packet loss rate of the last VNF under MDC is nearly zero, while substantial amount of packets dropping occursat the other positions of current service chain. It is also reflected in Fig. 3.PDCO achieves effective resource scheduling for VNFs with various types,to minimize the cost of packets dropping, it jointly considers the packets arrival rate and positions of VNFs instead of being blind for VNFs' status like other competitors.

Efficient resource scheduling scheme ensures the quality of network service providing and packet should be treated and processed in time.In the aspect of packet delay, it can be found from result in Fig. 5 thatthe packet delay is guaranteed underthe proposed PDCO algorithm, but it doesn't revealobvious advantage compared with others.

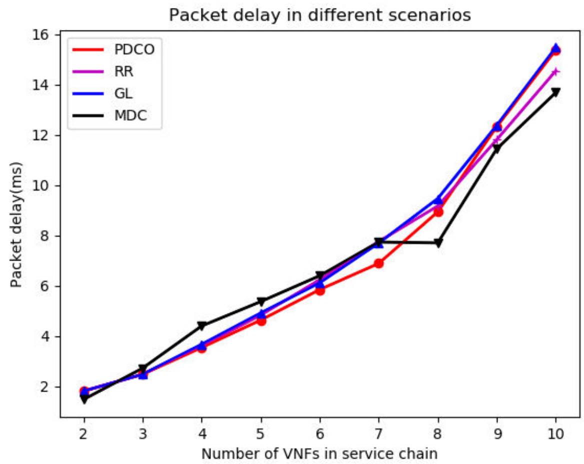

Fig. 5 Packet delay in different scenarios

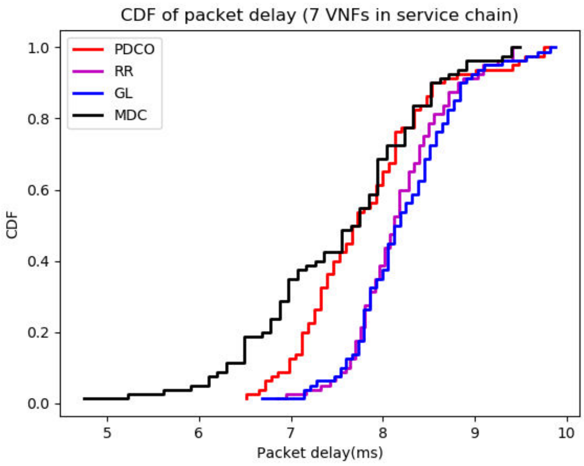

Fig. 6 CDF of packet delay

However, more details about packet delay have shown in specific circumstance, Fig. 6shows the CDF of single packet delaytraversing the whole service chainunder network scale withseven VNFs.Obviously,the PDCOoutperforms RR and GL overall. Althoughthe numberofpacketswhosedelayare lower than7ms is more under MDC, resource scheduling underPDCO also satisfies the traffic bandwidth requirement and avoidsseverecongestion.

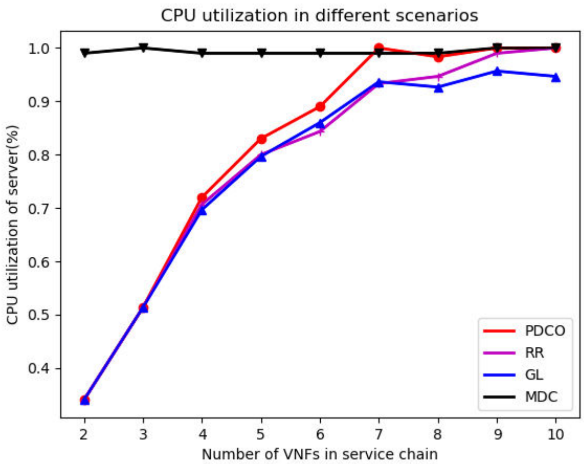

Fig. 7 CPU utilization in different scenarios

In addition, we also studiedCPU utilization of all algorithms under different network scales with the number of VNFs in service chain ranging from 2 to10.Because of constraint on computing resource, MDCalways utilizes totalCPU resource current server node has to conduct scheduling, but actually, the resource VNFs used in effect isless than it allocated due to taking no account ofactual packetarrival ratesto which PDCO pays attention. On the other hand, the performance of other two schemes in terms of CPU utilization is not as good asPDCO's as we can see in Fig. 7.

VI. CONCLUSION

Network Function virtualization is a promising design paradigm to efficiently manage network services. When multiple VNFs of service chains are deployed, losing packets at different positions would lead to various degree of resource cost due to wastage of upstream processing. In this paper, we focus on this issue and a greedy-based scheduling scheme is proposed as a solution to deal with resource scheduling problem in service chain environment.The comparison withexistingmethods in evaluation hasdemonstratedhigh effectiveness and optimality of PDCO scheduling algorithm we proposed.

REFERENCES

[1] Yi B, Wang X, Li K, et al. A comprehensive survey of network function virtualization[J]. Computer Networks, 2018, 133: 212-262.
[2] Kulkarni S G, Zhang W, Hwang J, et al. Nfvnice: Dynamic backpressure and scheduling for nfv service chains[C]//Proceedings of the Conference of the ACM Special Interest Group on Data Communication. 2017: 71-84.
[3] Qu L, Assi C, Shaban K. Delay-aware scheduling and resource optimization with network function virtualization[J]. IEEE Transactions on Communications, 2016, 64(9): 3746-3758.
[4] Pham C, Tran N H, Hong C S. Virtual network function scheduling: A matching game approach[J]. IEEE Communications Letters, 2017, 22(1): 69-72.
[5] Zhang Y, He F, Sato T, et al. Optimization of network service scheduling with resource sharing and preemption[C]//2019 IEEE 20th International Conference on High Performance Switching and Routing (HPSR). IEEE, 2019: 1-6.
[6] Mijumbi R, Serrat J, Gorricho J L, et al. Design and evaluation of algorithms for mapping and scheduling of virtual network functions[C]//Proceedings of the 2015 1st IEEE conference on network softwarization (NetSoft). IEEE, 2015: 1-9.
[7] Yao H, Xiong M, Li H, et al. Joint optimization of function mapping and preemptive scheduling for service chains in network function virtualization[J]. Future Generation Computer Systems, 2020, 108: 1112-1118.
[8] Alameddine H A, Sebbah S, Assi C. On the interplay between network function mapping and scheduling in VNF-based networks: A column generation approach[J]. IEEE Transactions on Network and Service Management, 2017, 14(4): 860-874.
[9] Yoshida M, Shen W, Kawabata T, et al. MORSA: A multi-objective resource scheduling algorithm for NFV infrastructure[C]//The 16th asia-pacific network operations and management symposium. IEEE, 2014: 1-6.
[10] Beck M T, Botero J F. Scalable and coordinated allocation of service function chains[J]. Computer Communications, 2017, 102: 78-88.
[11] Yi B, Wang X, Huang M. A generalized VNF sharing approach for service scheduling[J]. IEEE Communications Letters, 2017, 22(1): 73-76.
[12] El Khoury N, Ayoubi S, Assi C. Energy-aware placement and scheduling of network traffic flows with deadlines on virtual network functions[C]//2016 5th IEEE International Conference on Cloud Networking (Cloudnet). IEEE, 2016: 89-94.
[13] Li C, Cui L. A novel NFV schedule optimization approach with sensitivity to packets dropping positions[C]//Proceedings of the 2018 Workshop on Theory and Practice for Integrated Cloud, Fog and Edge Computing Paradigms. 2018: 23-28.
[14] Cui L, Tso F P, Guo S, et al. Enabling heterogeneous network function chaining[J]. IEEE Transactions on Parallel and Distributed Systems, 2018, 30(4): 842-854.
[15] Herrera J G, Botero J F. Resource allocation in NFV: A comprehensive survey[J]. IEEE Transactions on Network and Service Management, 2016, 13(3): 518-532.
[16] Labrador M A, Banerjee S. Packet dropping policies for ATM and IP networks[J]. IEEE Communications Surveys, 1999, 2(3): 2-14.
[17] Shortle J F, Thompson J M, Gross D, et al. Fundamentals of queueing theory[M]. John Wiley & Sons, 2018.
[18] Cidon I, Khamisy A, Sidi M. Analysis of packet loss processes in high-speed networks[J]. IEEE Transactions on Information Theory, 1993, 39(1): 98-108.
[19] Zhang Q, Xiao Y, Liu F, et al. Joint optimization of chain placement and request scheduling for network function virtualization[C]//2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS). IEEE, 2017: 731-741.

[20] Daigle J, Langford J. Models for analysis of packet voice communications systems[J]. IEEE Journal on selected areas in communications, 1986, 4(6): 847-855.

[21] Salvador P, Pacheco A, Valadas R. Modeling IP traffic: joint characterization of packet arrivals and packet sizes using BMAPs[J]. Computer Networks, 2004, 44(3): 335-352.

[22] Bolot J C. Characterizing end-to-end packet delay and loss in the internet[J]. Journal of High Speed Networks, 1993, 2(3): 305-323.

[23] Wang T, Liu F, Guo J, et al. Dynamic SDN controller assignment in data center networks: Stable matching with transfers[C]//IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications. IEEE, 2016: 1-9.

[24] Rasmussen R V, Trick M A. Round robin scheduling–a survey[J]. European Journal of Operational Research, 2008, 188(3): 617-636.