PAM: When Overloaded, Push Your Neighbor Aside!

Zili Meng[†], Jun Bi[#], Chen Sun[†], Shuhe Wang[†], Minhu Wang[†], Hongxin Hu[§]

[†#]Institute for Network Sciences and Cyberspace, Tsinghua University
[†#]Beijing National Research Center for Information Science and Technology (BNRist)
[†]{mengzl15, c-sun14, wang-sh14, wang-mh15}@mails.tsinghua.edu.cn, [#]junbi@tsinghua.edu.cn
[§]School of Computing, Clemson University, hongxih@clemson.edu

CCS CONCEPTS

• **Networks** → **Network dynamics**; *Network management*; Middle boxes / network appliances;

KEYWORDS

NFV; Service Chain; SmartNIC; Dynamic Scaling

ACM Reference format:
Zili Meng, Jun Bi, Chen Sun, *et al.*. 2018. PAM: When Overloaded, Push Your Neighbor Aside!. In *Proceedings of ACM SIGCOMM 2018 Conference Posters and Demos, Budapest, Hungary, August 20–25, 2018 (SIGCOMM Posters and Demos '18),* 3 pages.
https://doi.org/10.1145/3234200.3234215

1 INTRODUCTION

Network Function Virtualization (NFV) enables efficient development and management of network functions (NFs) by replacing dedicated middleboxes with virtualized Network Functions (vNFs). When a vNF is overloaded, network operators can easily scale it out by creating a new vNF instance and balancing the load between two instances. Meanwhile, network operators usually require packets to be processed by multiple vNFs in a certain sequence, which is referred to as a service chain [3]. However, the introduction of NFV results in *high latency*. Virtualization techniques in NFV significantly increase processing latency [7]. To address this problem, many research efforts from both industry [6] and academic communities [4] introduce programmable Network Interface Cards based on Network Processing Units (NPUs), *i.e.* SmartNICs, to accelerate NFV. With the advantage of high performance and resource efficiency, offloading vNFs from CPU to SmartNIC brings significant performance benefits.

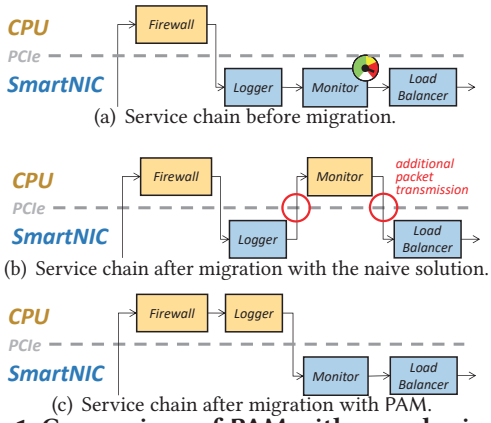

Figure 1: Comparison of PAM with casual migration. The service chain is derived from [7].

Meanwhile, as the network traffic fluctuates, NFs on SmartNIC can also be overloaded [4]. If we naively refer to the scaling out solutions for CPU, we have to introduce one more SmartNIC to alleviate the hot spot, which is hardly possible since each server is usually equipped with one or two SmartNICs only. UNO [4] proposed to alleviate the overload situation by identifying the *bottleneck vNF* with minimum processing capacity and migrating it to CPU. However, this intuitive naive solution may increase the latency of the service chain. As shown in Figure 1(b), if *Monitor* is the bottleneck vNF and we migrate it to CPU, packets have to be transmitted over PCIe for two more times. This adds tens of microseconds latency according to our experiments, which may be unacceptable for latency-sensitive applications [7].

To address this problem, in this poster, we propose PAM, the Push Aside Migration scheme, which identifies the *right vNFs* to migrate to alleviate the hot spot on SmartNIC without introducing long-term performance degradation. We consider from the scope of the *entire service chain* and propose our *key observation* that when a vNF is overloaded, we can migrate *other* vNFs on the SmartNIC away to *release resources* for the overloaded vNF. To avoid introducing extra packet transmissions over PCIe, we choose to migrate vNFs on the *border* of SmartNIC and CPU. As shown in Figure 1(c),

This poster is supported by the National Key R&D Program of China (No.2017YFB0801701), the National Science Foundation of China (No.61472213), the Tsinghua Initiative Research Program (No.20181080342), and the National Training Program of Innovation and Entrepreneurship for Undergraduates. Jun Bi is the corresponding author.

SIGCOMM Posters and Demos '18, August 20–25, 2018, Budapest, Hungary

ACM ISBN 978-1-4503-5915-3/18/08…$15.00
https://doi.org/10.1145/3234200.3234215

Table 1: Capacity of vNFs on the SmartNIC and CPU.

vNF i	Firewall	Logger	Monitor	Load Balancer
θ_i^S	10 Gbps	2 Gbps	3.2 Gbps	>10 Gbps
θ_i^C	4 Gbps	4 Gbps	10 Gbps	4 Gbps

we migrate the *Logger* to CPU to alleviate the *Monitor* hot spot. However, selecting the right border vNFs for migration is challenging. Migrating too few vNFs may not effectively alleviate the hot spot, while migrating too many vNFs may waste CPU resource. To address this challenge, PAM carefully models SmartNIC and CPU resources and proposes an effective algorithm to find the most suitable vNFs for migration. Our evaluation shows that PAM could effectively alleviate the hot spot on SmartNIC and generate a service chain with 18% lower latency compared with the naive solution.

2 PAM DESIGN

In this section, we first introduce the resource constraints of SmartNIC and CPU. We then introduce how PAM identifies proper border elements on SmartNIC for migration to effectively alleviate the hot spots on SmartNIC without performance degradation due to extra packet transmissions.

To understand the resource constraints of the CPU and SmartNIC, we refer to [5] and assume that *the resource utilization of a vNF on both SmartNIC and CPU increases linearly with the vNF throughput.* Suppose the throughput capacity of vNF i on SmartNIC is θ_i^S and the current throughput is θ_{cur}, the ratio of consumed resource on SmartNIC is θ_{cur}/θ_i^S. We measure and present the capacity of several vNFs in Table 1. We adopt the NF migration mechanism between SmartNIC and CPU introduced in [4]. The network administrators can periodically query the load of SmartNIC and CPU and execute the PAM border vNF selection algorithm:

Step 1: Border vNFs Identification. We first find out the border vNFs of SmartNIC. We classify them into left border and right border vNFs, whose upstream or downstream vNF is placed on CPU. For example, the left border vNF in Figure 1(a) is *Logger* and the right border vNF is *Firewall.* Due to the several packet transmissions between SmartNIC and CPU, there may be multiple border vNFs in a service chain. We respectively denote them as set $\mathcal{B}_L$ and $\mathcal{B}_R$. *Migrating border vNFs will not introduce new packet transmissions.*

Step 2: Migration vNF Selection. To alleviate the overload with minimum number of vNF to migrate, we select the vNF b_0 from border vNFs with minimum capacity on SmartNIC:

$$b_0 = \underset{b \in \mathcal{B}_L \cup \mathcal{B}_R}{\arg\min}\ \theta_b^S \tag{1}$$

Step 3: Overload Alleviation Check. Meanwhile, we need to ensure ① migration will not cause new hot spots on CPU, and ② the overload of SmartNIC can be alleviated. For ①:

$$\sum_{i \in \{vNFs\ on\ C\}} \frac{\theta_{cur}}{\theta_i^C} + \frac{\theta_{cur}}{\theta_{b_0}^C} < 1 \tag{2}$$

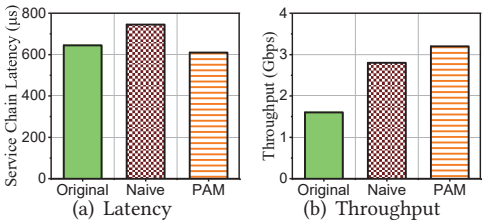

Figure 2: Comparison of the naive solution and PAM.

If Equation 2 is not satisfied, which indicates migration will create new hot spots on CPU, we cannot migrate it to CPU. We remove b_0 from $\mathcal{B}_L$ or $\mathcal{B}_R$ and go back to Step 2. Otherwise, we can continue to check constraint ②:

$$\sum_{i \in \{vNFs\ on\ S\},\ i \neq b_0} \frac{\theta_{cur}}{\theta_i^S} < 1 \tag{3}$$

The algorithm terminates if Equation 3 is satisfied. Otherwise, we migrate b_0 to CPU. If $b_0 \in \mathcal{B}_L$, we remove it from $\mathcal{B}_L$ and add its downstream element into the set if the downstream element is also placed on SmartNIC. If $b_0 \in \mathcal{B}_R$, we execute similar actions on its upstream element. We then go back to Step 2 to continue the loop. If both CPU and SmartNIC are overloaded, which rarely happens, the network operator must start another instance to alleviate the hot spot [1].

3 PRELIMINARY EVALUATION

We implement the service chain in Figure 1 on a server equipped with one Netronome Agilio CX 2×10GbE SmartNIC [6], two Intel Xeon E5-2620 v2 CPUs (2.10 GHz, 6 physical cores), and 128G RAM. For the naive algorithm, we pick the vNF on SmartNIC with minimal capacity θ_{NF}^S. We measure the service chain throughput and latency of different migration selection mechanisms in Figure 1. We vary the packet size from 64B to 1500B with a DPDK packet sender [2] and present the average latency and throughput in Figure 2.

PAM decreases the service chain latency by 18% on average compared to the naive solution. The service chain latency with PAM is almost unchanged compared to the latency before migration because PAM does not introduce redundant packet transmissions. Meanwhile, the throughput of the service chain of PAM is improved a little since NFs may perform differently on SmartNIC and CPU.

4 CONCLUSION AND FUTURE WORK

We have proposed a vNF selection scheme, PAM, which reduces the *service chain latency* when alleviating hot spots on SmartNIC. We present our key novelty of pushing the *border vNFs* aside to release resources for the *bottleneck vNF.* Evaluation shows that PAM could alleviate the hot spot on SmartNIC and generate a service chain with 18% lower latency compared with the naive solution. As our future work, we will analyze PCIe transmissions in detail, consider the difference of processing the same vNF on both devices, and extend PAM to work in FPGA-based SmartNICs.

REFERENCES

[1] Aaron Gember-Jacobson, Raajay Viswanathan, Chaithan Prakash, Robert Grandl, Junaid Khalid, Sourav Das, and Aditya Akella. 2014. OpenNF: Enabling innovation in network function control. In *Proceedings of the Conference of the ACM Special Interest Group on Data Communication (SIGCOMM'14)*. ACM, 163–174.

[2] DPDK Intel. 2018. Data Plane Development Kit. (2018). http://dpdk.org

[3] S Kumar, M Tufail, S Majee, C Captari, and S Homma. 2015. Service function chaining use cases in data centers. *IETF SFC WG* (2015).

[4] Yanfang Le, Hyunseok Chang, Sarit Mukherjee, Limin Wang, Aditya Akella, Michael M Swift, and TV Lakshman. 2017. UNO: uniflying host and smart NIC offload for flexible packet processing. In *Proceedings of the 2017 Symposium on Cloud Computing (SoCC'17)*. ACM, 506–519.

[5] Zili Meng, Jun Bi, Chen Sun, Haiping Wang, and Hongxin Hu. 2018. CoCo: Compact and Optimized Consolidation of Modularized Service Function Chains in NFV. In *Proceedings of the 2018 International Conference on Communications (ICC'18)*. IEEE.

[6] Netronome. 2018. Netronome Agilio CX Dual-Port 10 Gigabit Ethernet SmartNIC. (2018). http://www.colfaxdirect.com/store/pc/viewPrd.asp?idproduct=3017

[7] Chen Sun, Jun Bi, Zhilong Zheng, Heng Yu, and Hongxin Hu. 2017. NFP: Enabling Network Function Parallelism in NFV. In *Proceedings of the Conference of the ACM Special Interest Group on Data Communication (SIGCOMM'17)*. ACM, 43–56.