Traffic Load Balancing based on Probabilistic Routing in Data Center Networks

Fu Wang
Department of Electronic Engineering
Eindhoven University of Technology
Eindhoven, Netherland
f.wang2@tue.nl

Fulong Yan
Department of Electronic Engineering
Eindhoven University of Technology
Eindhoven, Netherland
f.yan@tue.nl

Xuwei Xue
Department of Electronic Engineering
Eindhoven University of Technology
Eindhoven, Netherlandline
x.xue.1@tue.nl

Bo Liu
School of physics and optoelectronic engineering
Nanjing University of Information Science and Technology
Nanjing, China
bo@nuist.edu.cn

Lijia Zhang
School of Electronic Engineering
Beijing University of Posts and Telecommunications
Beijing, China
zlj@bupt.edu.cn

Qi Zhang
School of Electronic Engineering
Beijing University of Posts and Telecommunications
Beijing, China
zhangqi@bupt.edu.cn

Xiangjun Xin
School of Electronic Engineering
Beijing University of Posts and Telecommunications
Beijing, China
xjxin@bupt.edu.cn

Nicola Calabretta
Department of Electronic Engineering
Eindhoven University of Technology
Eindhoven, Netherlandline
n.calabretta@tue.nl

***Abstract*—We investigate a novel load-balancing algorithm exploiting probabilistic dynamic routing for OPSquare DCN. Results show that for 10240-server DCN, the proposed load-balancing decreases the ToR-to-ToR latency by 27.3% and packet loss by 56.45% at a load of 0.6.**

Keywords—data center networking, load balancing, OPSquare, probabilistic routing

I. INTRODUCTION

Facing the diverse applications with different requirements on latency and bandwidth, such as 5G and flow media, the traffic load in DCN is unbalanced, causing link congestion and leading to packet loss [1, 2]. The load balance strategy is crucial to reduce the packet loss and latency in large-scale DCN. The load balance includes two aspects, server-level balancing and traffic-level balancing. Server-level balancing mainly involves Virtual Machine (VM) migration [3]. VM migration tries to transfer traffic generators to decrease hot-point traffic in DCN, whereas traffic-level balancing changes the routing path of generated traffic to decrease the link congestion. The former would introduce extra traffic, even cause service disruption, especially at high load, when servers migrate VMs. The latter is a cost-effective way for load balancing, and how to arrange the multiple paths for packets in ToRs is still a hot topic. Equal-Cost Multi-Path (ECMP) is a typical approach based on a per-flow static hashing [4]. This algorithm randomly splits flows into the multiple equal-cost paths without leveraging the link state information or buffer occupancy information. Some advanced algorithm based on ECMP, like TFE, has been proposed for Ethernet switched networks [5]. However, with the deployment of Fast Optical Switches (FOS), hybrid electrical/optical DCN with different bandwidths increases the complexity of the load balancing. A good DCN architecture helps to improve the DCN performance without increasing the costs. An efficient load balancing algorithm can also improve DCN performance. OPSquare DCN architecture [6,7] can provide any cluster-to-cluster connection with at most two hops. It could be an efficient architecture to balance the traffic load due to the availability of multiple paths between the Top-Of-Racks (TOR). However, there are no studies/investigations of a dedicated load balancing algorithm to efficiently exploit the multiple paths.

In this work, we propose a novel algorithm called Load Balancing based on Probabilistic dynamic Routing (LBPR) for multi-path arrangement. We adopt a dynamic probability of the path arrangement. When a packet enters into a TOR switch, there are multiple paths for the next hop. The packet would match one of the paths based on the hashing. In the proposed algorithm, the forwarding probability is dynamically adjusted based on the buffer occupancy information in the ToR transceivers (TRX). Numerical simulation indicates that the LBPR decreases the ToR-to-ToR latency by 27.3% and packet loss by 56.45% at a load of 0.6 with respect of ECMP.

II. PRINCIPLE OF OPERATION OF THE TRAFFIC LOAD BALANCING BASED ON PROBABILISTIC ROUTING

A. OPSquare DCN based on optical switching

Fig. 1(a) shows the OPSquare architecture based on FOS. This DCN consists of ToR switches, intra-cluster FOSs (ISs), and inter-cluster FOS (ESs). ToR switches provide inter-server connectivity and multiple high-speed optical interfaces for ISs and ESs (see Fig 1(b)). The ISs are responsible for inter-cluster traffic. The ToRs in a cluster are divided into multi-groups. A TRX is responsible for the traffic within a group. Packets with the same group would be aggregated to the same TRX buffer. ESs connect the same indexed ToRs in all clusters, as shown in Fig 1(a). The OPSquare DCN architecture can provide two equal-cost paths for inter-cluster traffic. For example, ToR(n*m-m+1) in p-th cluster can be connected to ToR1 in 1st cluster by

978-3-903176-21-8

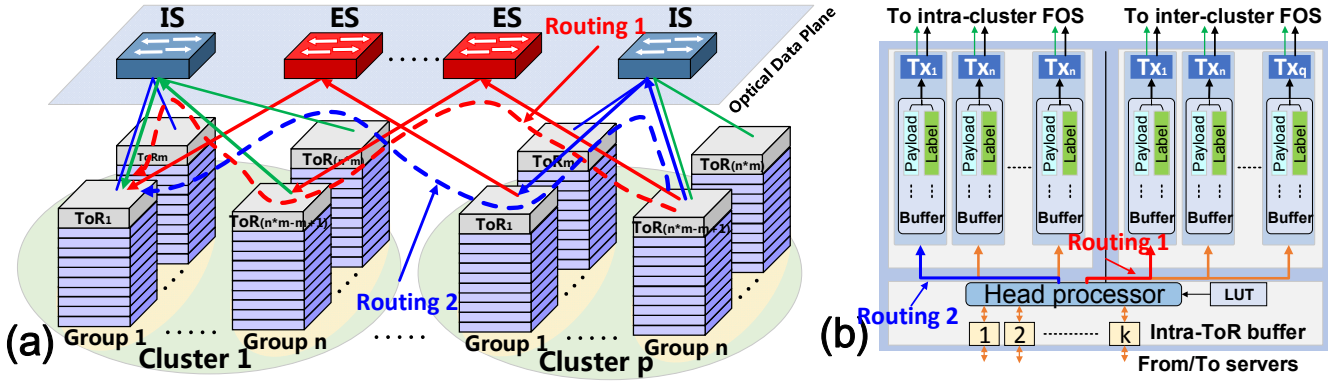

Fig. 1. (a)OPSquare DCN based on FOS (b) ToR with multi-transmitters.

two possible connections, as the red dotted line and blue dotted line shown in Fig. 1(a). The existing electrical switches adopt ECMP in ToRs to arrange the routing paths. As shown in Fig. 1(b), the head processor of each ToR handles the Ethernet frames coming from the servers based on a hashing key. The head processor calculates the key from the frame header information. The key space, seen as flow table, includes an entry list to match the key following with an action (deliver to the output or drop). Typically, each entry has a priority number to indicate the matching order (high priority will be matched firstly). After matching the entry in key space, the head processor forwards the Ethernet frame to an buffer of a TRX to be transmitted. This matching scheme can provide a fixed routing path for each flow.

ECMP averagely distributes the traffic flow into equal-cost paths without any information from the buffer occupancy or link-state even if one of the equal-cost paths is congested. If a TOR switch adds or removes a path for a key, the TOR need to modify a flow entry (add for remove) in the flow table. Flow means a continuous series of Ethernet frames with the same header in the MAC layer. The key space is distributed into multi equal-cost paths averagely. If we add or remove any equal-cost path into the flow table, the key space will be redistributed to the remaining equal-cost paths.

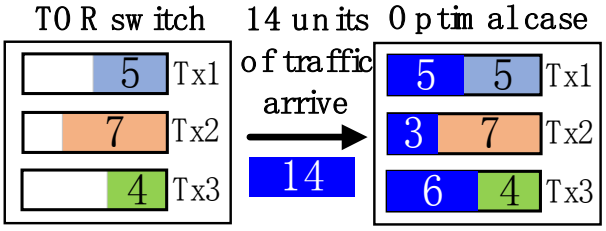

Fig. 2. The optimal case for LBPR in TOR.

B. *Load balance based on probabilistic dynamic routing*

Facing with the rapidly changing of the traffic of the DCN, it is challenging to realize the real-time management of the flow table in SDN-enable DCN. It usually takes tens of milliseconds to update the flow table. Therefore, we propose an LBPR to implement the traffic load balancing. The flow table of LBPR includes a set of keys and actions to indicate the different paths for a specific flow. Each action corresponds to a key. The Ethernet frames with same header involve the same key in the switch. When a flow matches one flow entry in the flow table, the flow will execute the action of the key to be forwarded.

The size of a region in LBPR indicates the probability to be matched. The sum of the probabilities in the flow table is 100% (all key space). Furthermore, the distribution of key regions is updated periodically at an interval. After a period of each interval, the ToRs check the occupancy state of the TRX buffers and then update the proportion of regions in key space according to (1).

$$P_{act}(a)=\frac{p_{min}}{n}+(1-p_{min})*\frac{(B_{max}-B(a)+\mu)}{\sum_{i=1}^{n}(B_{max}-B(i)+\mu)} \quad (1)$$

$P_{act}(a)$ is the proportion of region (a) in the flow table. $B(a)$ is the buffer occupancy volume of TRX(a) and B_{max} is the buffer size (all buffers have the same size), so B_{max}-$B(a)$ is the free space of $B(a)$. μ is a value to prevent equation exceptions when all buffers are full. We assume that there are n TRXs (possible equal-cost paths, from i=1...n) in the flow table. $B(i)$ represents the buffer occupancy of the i-th path in the entry (from $B(1)$ to $B(n)$). We suppose that the Ethernet frames have n actions (means n paths). We set a minimum probability (p_{min}) for a path. When the buffer of TX is full, the action for this TRX still has a probability of p_{min}/n. As illustrated in Fig. 2, an example shows the optimal case for the load balancing. The optimal object for load balancing algorithm is trying to distribute the traffic into different buffers of equal-cost paths to reduce packet loss. When the incoming traffic volume is the same as the sum of all remaining buffer, all buffers should be full without packet loss. Equation (1) can achieve better load balancing probabilistically. If we calculate the proportion of region via free space of all buffers, the load balance will reach the optimal object. However, we cannot adjust the proportion of all regions for each Ethernet frame in real time. Therefore, we recompute the proportion of the regions by a certain time interval. For a buffer of 51.2 KB, the ratio distribution of two paths changes nonlinearly. When the occupancies of two buffers are same, the proportions is also same for two paths.

III. Numerical Results Analysis

A numerical simulation is performed for 10240-server OPSquare DCN scenarios by OMNET++ to validate the LBPR algorithm shown in Fig. 1(a). The DCN consists of 16 clusters,

The authors would like to thank the European Union's Horizon 2020 research and innovation programme under grant agreement PASSION No 780326 and the EU H2020 QAMELEON project (n° 780354) for partially supporting this work.

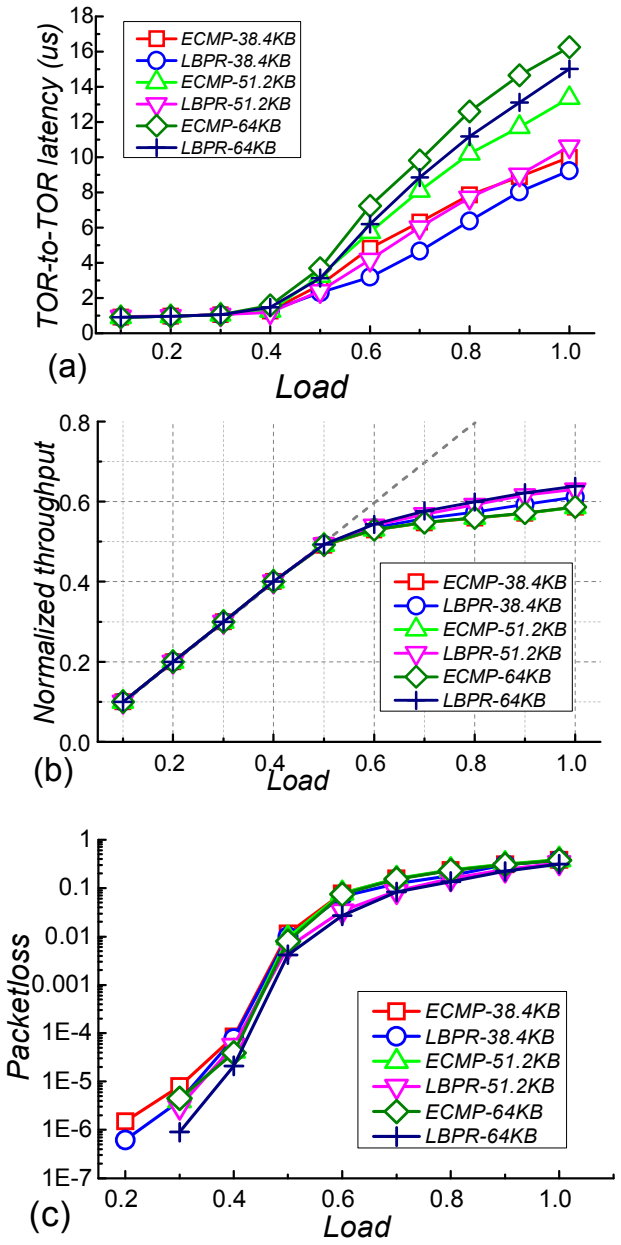

Fig. 3. (a) ToR-to-ToR latency vs. load. (b) Packet loss vs. load. (c) Throughput vs. load.

and each cluster contains 16 ToRs. Each ToR interconnects 40 servers. An ON-OFF model is used to generate Ethernet frames. The size of Ethernet frames follows a bimodal distribution with peaks around 64 B and 1518 B. The size of traffic flow follows Generalized Pareto Distribution (GPD). The ToR is equipped with inter-\intra-cluster TRXs at 40 G, while the TRXs of the servers operate at 10G (400 G traffic in the TOR). The ON-OFF traffic model can control the load from 0.1 to 1. The inter-ToR traffic is 50% of total traffic. The intra-cluster and inter-cluster traffic are 40% and 10% of the total traffic. There are four TRXs for the intra-cluster traffic (total 160 G bandwidth). Only one TRX (40 G) is used for inter-cluster traffic. "Load of 1.0" means the OFF period of servers is 0. The propagation delay between ToRs and FOS is 250 ns. The propagation latency between the server and the ToR is 15 ns. Each TRX is equipped with a buffer between 51.2 KB size. The interval time for updating the LBPR key space is 1us. The inter/intra switches are FOSs with 900 ns packet length, 50ns label processing time, and 50 ns gap time between packets. The ToR-to-ToR connection includes two equal-cost paths. p_{min} is set to 0.2, and μ is set to 6.4 KB.

Fig. 3(a) shows the latency performance. Two algorithms perform similarly at low load <0.4 since the LBPR has little impact on routing probability at low load (low buffer occupancy). For load >0.4, the LBPR performs better. For the buffer size from 38.4 KB to 64 KB, LBPR shows excellent performance compared with ECMP. Numerical simulation indicates that the LBPR decreases the ToR-to-ToR latency by 27.3% at the load of 0.6.Fig. 3(b) shows that LBPR performs better than ECMP in throughput. Compared with ECMP, LBPR improve throughput performance when the load is higher than 0.5. ECMP shows almost the same performance for different buffer sizes at load>0.5. The LBPR increase throughput of 4.77% at a load of 0.6 (51.2 KB) compared with ECMP. Fig. 3(c) shows the packet loss performance for ECMP and LBPR. LBPR can decrease the packet loss effectively, especially for high load case. The LBPR decreases packet loss by 56.45% compared with ECMP at a load of 0.6 (51.2 KB).

IV. Conculsion

We have proposed and investigated a novel load balancing scheme based on probability routing (LBPR). The LBPR algorithm can distribute traffic to idle paths by the dynamic probability to prevent traffic congestion. A simulation for 10240-server DCN was carried out to validate its performance. The proposed algorithm improves the system performance concerning ECMP by decreasing the ToR-to-ToR latency of 27.3% and the packet loss of 56.45% at the load of 0.6 (51.2 KB).

Acknowledgment

The authors would like to thank the European Union's Horizon 2020 research and innovation programme under grant agreement PASSION No 780326 for partially supporting this work.

References

[1] "Cisco Global Cloud Index: Forecast and Methodology", 2016–2021 White Paper.

[2] J. Zheng, et al, "Dynamic Load Balancing in Hybrid Switching Data Center Networks with Converters," in ICPP, (ACM, Kyoto, 2019).

[3] L. Yu, L. Chen, Z. Cai, et al, "Stochastic Load Balancing for Virtual Resource Management in Datacenters," in IEEE Transactions on Cloud Computing.

[4] M. Chiesa et al, "Traffic Engineering With Equal-Cost-Multi Path: An Algorithmic Perspective," IEEE T. Networking, vol. 25, pp. 779–792, 2017.

[5] J. Alvarez-Horcajo, D. Lopez-Pajares, I. Martinez-Yelmo, J. A. Carral and J. M. Arco, "Improving Multipath Routing of TCP Flows by Network Exploration," in IEEE Access, vol. 7, pp. 13608-13621, 2019.

[6] X. Xue et al., "SDN-Controlled and Orchestrated OPSquare DCN Enabling Automatic Network Slicing With Differentiated QoS Provisioning," in Journal of Lightwave Technology, vol. 38, no. 6, pp. 1103-1112, 15 March15, 2020.

[7] F. Wang et al., "Demonstration of SDN-enabled Hybrid Polling Algorithm for Packet Contention Resolution in Optical Data Center Network," in Journal of Lightwave Technology.