Energy-Efficient VNF Replication in Virtualized Data Centers

Janani Janardhanan, Bin Tang, Mohsen Beheshti
jjanardhanan1@toromail.csudh.edu, btang, mbeheshti@csudh.edu
Computer Science Department, California State University Dominguez Hills

Abstract

A Virtual Network Function (VNF) refers to the implementation of a network function, such as Firewall, Load Balancer, Network Address Translator (NAT), and Intrusion Detection System (IDS) using software decoupled from the underlying hardware. These network functions, also referred to as middle boxes, primarily ensure secure and cost-effective traffic flow in Virtualized Data Center Networks. . The efficient distribution of these VNFs directly impact network security and performance. Although some studies have been conducted on optimal placement of VNFs, very few of them considered the replication of VNFs in the network for minimizing cost flow, addressing load balancing and fault-tolerance issues. This research proposes two algorithms namely Closest Next Middlebox First and Traffic-aware VNF Replication to accomplish energy efficient and cost-effective replication by distributing copies of VNFs based on the expected traffic flow within the virtual machines. We also show via extensive simulations that both the algorithms outperform random replication techniques and that traffic-aware algorithm performs better for fat-tree network.

Introduction

Virtual Network Function transforms, inspects, filters, or otherwise manipulates traffic for purposes other than packet forwarding. Individual virtual network functions can be chained together as building blocks to offer a full-scale networking communication service. This is called service-chaining. Although some studies have been conducted on optimal placement of service chains, very few of them considered the replication of VNFs in the network for minimizing cost flow, addressing load balancing and fault-tolerance issues. The Closest Next Middlebox First (CNMF) algorithm works based on the physical proximity of the communicating Virtual Machine(VM) pairs and their corresponding subsequent network functions in a service chain. The Traffic-Aware VNF Replication (TAVR) primarily focuses on replicating VNFs by prioritizing their usage demands by various VM pairs in the network based on their communication frequencies.

Fat-Tree Data Center

A k-ary fat-tree is shown in Figure 1 with k = 4, where k is the number of ports of each switch. There are three layers of switches: edge switch, aggregation switch and core switch A fat-tree built with k-port switches supports k³/4 physical machines that can host Virtual Machines. Totally, there are 5k²/4 switches in the entire network. Initially, an original service chain is available on the network for further replication. An example service chain of length 5 is illustrated in this figure.

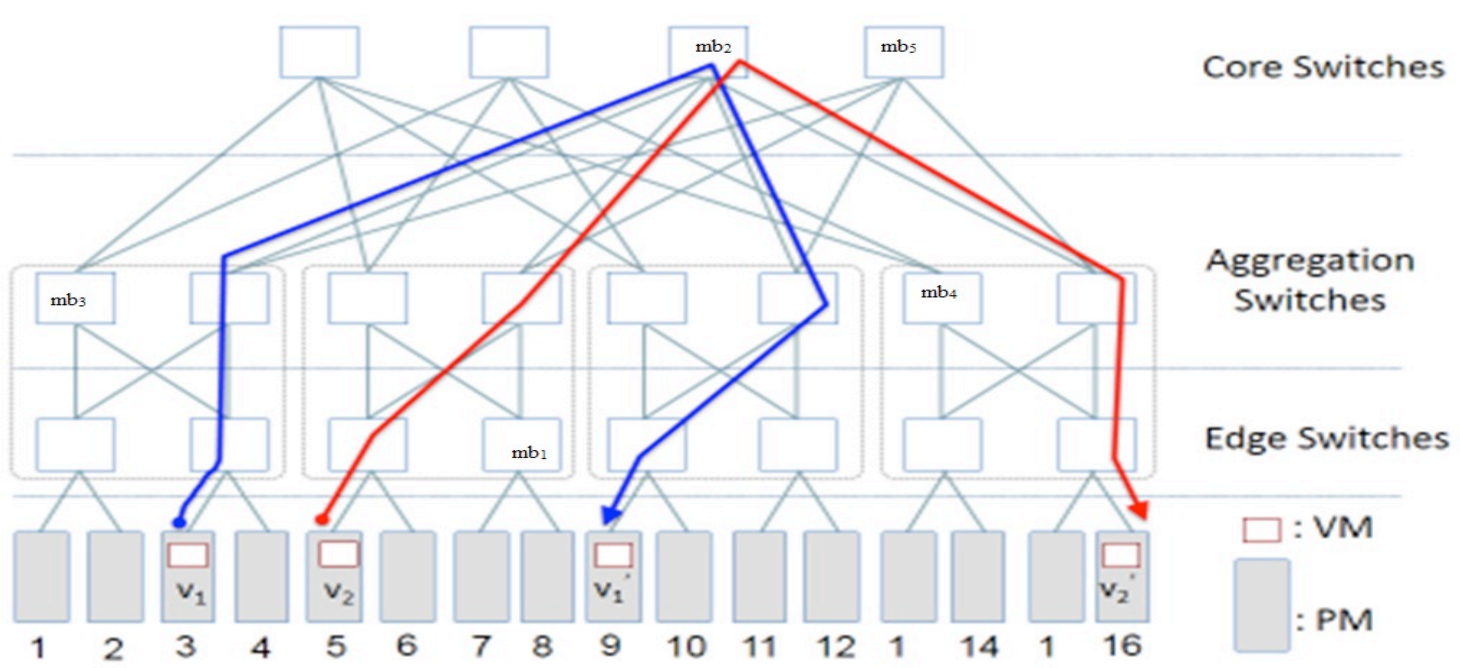


Figure 1. Sample Fat-Tree Topology

Problem Formulation

- Given a data center graph *G(V,E) (See Figure 1)*, that has m MBs $M=\{mb_1, mb_2, \ldots, mb_m\}$
- A set of *p* communicating node pairs *P*, each pair *(s,t)* in *P* needs to traverse $mb_1, mb_2, \ldots, mb_m$ in that order
- The cost for $p = (s,t)$ is $c(p) = d(s, mb_1) + d(mb_1, mb_2) + \ldots + d(mb_{m-1}, mb_m) + d(mb_m, t)$
- Goal: Effective replication of m MBs in data center such that there is high availability of middleboxes and overall network cost for traffic flow among all *p* pairs is minimized

Algorithms and Time Complexity

Closest Next Middlebox First Algorithm (CNMF):

- Takes place in $5k^2/4m$ rounds where $5k^2/4$ is the number of switches and m is the number of middlebox types.
- Every round places a replica copy of every middlebox type by considering every available switch as a host. If the current switch X gives the least cost for a given middlebox type, it is chosen as the destination.
- At every iteration, the service chains for all the VM pairs(say, P) are formed by choosing the next closest middlebox type in the sequence.
- Time complexity for replication is $(5k^2/4m)*m*(5k^2/4)*P = P|K|^4$, while for choosing the next closest middlebox: $(P * 5k^2/4m + m*5k^2/4m)$
- Overall time complexity for replication: $P^2|K|^6$

Traffic-Aware VNF Replication Algorithm (TAVR):

- The communicating VM pairs are labeled based on their communication frequency as one of {Very Frequent, Frequent, Medium, Rare} communicators and replication is done in favor of traffic groups in the order of highest frequency.
- The iterations are similar to CNMF but at every iteration, a service chain is completely replicated as long as it yields a traffic cost that is at least the same as the traffic cost yielded by original service chain.
- Time complexity for replication is $(5k^2/4m)*m*(5k^2/4) * P = P|K|^4$

Simulation Parameters and Analysis

- *K = 4 and 8*
- *P = 100, 200, 300, 400, and 500*
- *m = 3, 5 and 7.*
- A random replication algorithm in which middleboxes are randomly replicated and an exhaustive middlebox replication algorithm in which all combinations are tried out before replicating a middlebox are designed to highlight relatively better performance of CNMF and TAVR algorithms.
- When k and p are fixed, with the increase of m, cost increases. With the increase of m, performance difference between CNMF and TAVR increases.
- Both the algorithms outperform random replication algorithm. While exhaustive MB replication is an ideal algorithm, it has a very long convergence time.

Figure 2

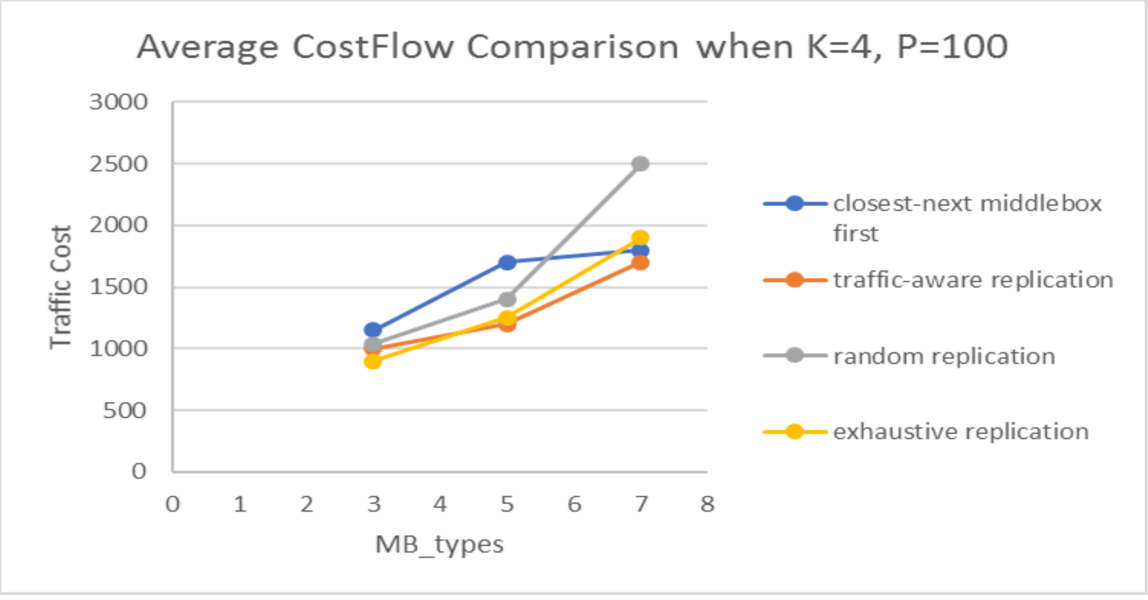
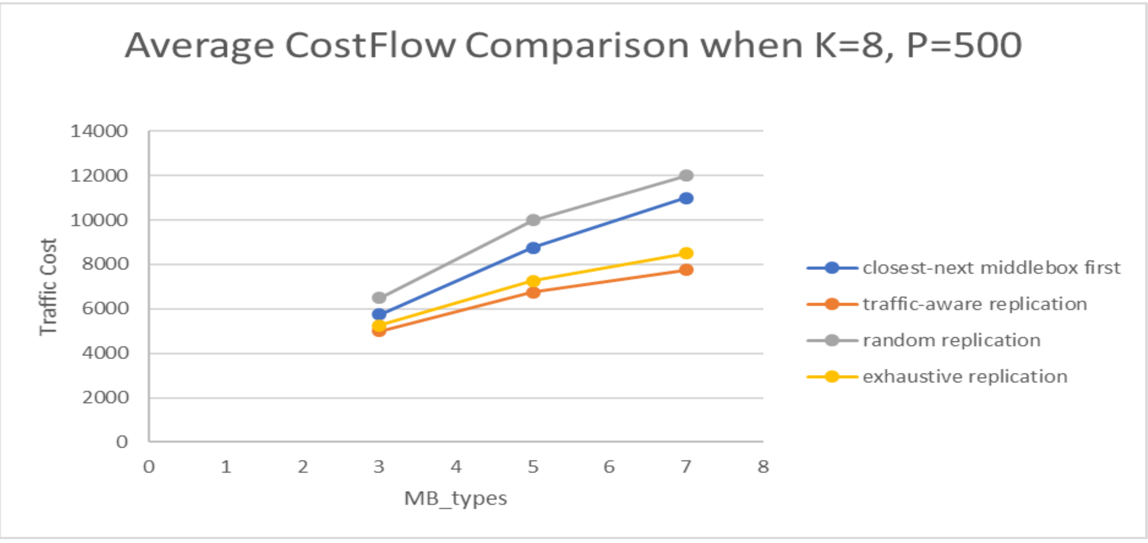


Figure 3



Conclusions

- We formulate energy-efficient and cost-effective middlebox/VNF replication problem and designed two algorithms for virtualized data centers.
- Both the algorithms are efficient as they have low convergence time and are easily scalable.
- Extensive simulations show that The Traffic-Aware VNF Replication algorithm outperforms Closest Next Middlebox First algorithm for large number of middleboxes approximately by 22%.

Future Work

- VNF Replication for special cases in which the order of the network functions to be performed on the traffic flow is random.
- Adapting the algorithms to perform efficient replication when different VM pairs have different service chain policies.

Acknowledgements