A Two-Phase Multipathing Scheme based on Genetic Algorithm for Data Center Networking

Lyno Henrique Gonçalvez Ferraz, Diogo Menezes Ferrazani Mattos,
Otto Carlos Muniz Bandeira Duarte
Grupo de Teleinformática e Automação
Universidade Federal do Rio de Janeiro (COPPE/UFRJ)
Rio de Janeiro – RJ – Brasil
Email:{lyno,menezes,otto}@gta.ufrj.br

***Abstract*—Data centers for cloud computing should allocate services with different traffic patterns, provide high data transfer capacity and link fault tolerance. Data center network topologies provide physical connection redundancy, which forwarding mechanisms avail to generate multiple paths. In this paper, we divide multipathing into two phases: (i) Configuration phase based on genetic algorithms to minimize path lengths and maximize link usage diversity; (ii) Path selection phase based on heuristics to minimize path reuse. The proposed multipathing scheme implements minimal modification in infrastructure. Our proposal only requires common network devices features and it avoids any tenant modification. We develop a flow simulator to evaluate multipathing techniques. The simulations model flow behaviors in different data center scenarios and compares the proposed scheme with multipathing techniques in literature. The results show the proposed scheme enhances transmission rates, even in the highest network utilization scenarios.**

I. Introduction

Data generation grows exponentially and several technologies, such as, the Internet of things and smart grid, worsen this scenario. The data storage and manipulation compose an area called Big Data, in which cloud computing and data centers perform crucial roles. The number of hosted applications in cloud data centers grows and, consequently, it demands high throughput between servers in a data center. The data center offers high processing and storage capacity by interconnecting several servers. The applications are distributed across the servers, hence the network plays a fundamental role to ensure applications to meet their needs [1].

Cloud providers host several applications of different clients composing a multi-tenant scenario, in which tenants share servers and network with each other [2]. As each tenant has its own applications and communication protocols, the cloud provider should offer high transmission rates, without changing in tenant systems, software or protocols. One way to improve data center network performance is using multipathing techniques. The data center network topologies usually have redundant paths, so they provide reliability and high aggregated bandwidth. The topologies are formed with multi-rooted trees to offer multiples paths between any pair of servers and forwarding mechanisms use the redundant paths mostly for failure tolerance [3]. Therefore, modifications to provide high transmission rates should be employed only in infrastructure, exploring available multipath. The multipathing scheme should keep modification in infrastructure in a minimum level to ensure easy deployability.

In this paper, we propose a multipathing scheme that divides the problem into two phases: Multipath Configuration and Multipath Selection. In Multipath Configuration phase, the scheme creates multipaths based on a genetic algorithm, whose objective function is to minimize the path sizes and to maximize link usage diversity. The multipaths use a common feature of data center switches, Virtual Local Area Network (VLAN) tag, to map multipaths into multiple trees. The genetic algorithm simplifies the mapping problem and allows the creation of VLAN trees with multiple objective functions, such as path sizes minimization and link usage diversity maximization. Furthermore, we model heuristics which the scheme uses to select one of the paths to an upcoming flow during the Multipath Selection phase. We also present cloud data center demands and evaluate multipathing techniques. The results compare the proposal with prominent multipathing techniques, such as, Spanning Tree Protocol [4], Equal Cost MultiPath [5], Smart Path Assignment In Networks (SPAIN) [6].

We developed a discrete-event flow-level simulator to analyze the performance of the multipathing techniques. The flow-level simulator allows greater scale simulations compared to packet-level simulations. The simulator uses a simplified flow-model to share the data center links bandwidth. Results show that the proposed scheme always achieves superior transmission rates, both in all-to-all and all-to-one scenarios.

The rest of the paper is structured as follow. Section II presents virtualized cloud data center issues. In Section III, we present the genetic algorithm for creating multipaths and the heuristics for selecting path. The developed simulator is described in Section IV and the results, in Section V. Finally, Section VI concludes the paper.

II. Comunication Systems in Virtualized Cloud Data Centers

The traditional data centers use dedicated servers to run specific applications, which result in inefficient resources utilization due to demand variations and consequently the unused resources [7]. The growth of cloud computing and virtualization technologies enables the aggregation of several services and applications in a single data center. The services and applications are virtual networks that share the data center resources [8]. The aggregation increases resource usage and reduces operation and maintenance costs. Besides, the virtualized cloud data centers host several tenants, each has own applications and services with different requirements. Therefore, the diversity of applications sharing the infrastructure cause in a variety of traffic patterns in data center network [1].

The data center topology is built with the goal of providing high computation capacity with several servers interconnected. To connect the computational resources, the network topology employs several paths between servers [9]. Thus, an application is distributed across servers, which results in a intra data center traffic four times bigger than the outgoing traffic [10].

The applications running in the data center are varied, each with distinct traffic pattern [11]. Most of the flows are small and short-lived, the so called mice flows. On the other hand, most of the bytes are transferred in big and long-lived flows, which compose a minority of total flows. The big and long-lived flows are the elephant flows. The mice and elephant flows have distinct requirements and behaviors, which the interaction between them causes packet losses and delays in requests fulfillment that deteriorate the applications performance.

The mice flows are latency-sensitive, because normally are generated by partition/aggregate applications, such as Map/Reduce applications used in web searches, social network content composition, and advertising selection [12]. In this type of application, the aggregator divides a request into sub-requests and delivers them to workers during the partition phase. After the computation of the sub-requests, the workers send the result to the aggregator, which in turn perform the aggregation phase. The flows of this kind of application have strict delay limits so the responses are timely presented to the users. The responses which violates the delay limits are discarded, which degrades the overall quality of the response to users. The main cause of the response delay increase is the flash congestion during the aggregation phase, which corresponds to a many-to-one communication. Hence, the prevention of the issue is the prioritization of delay-sensitive flows and low link usage to avoid traffic congestion losses [13]. The proposals in literature not only require several infrastructure devices modifications, but also modify end servers applications and protocols to interact with the infrastructure devices, normally to reserve resources [13], [12]. Thus, these proposals are not suited to cloud data centers with multi tenants, since each tenant has own distinct applications and protocols. Besides, even if the protocols are standardized, the interaction between the applications and network devices present security risks due to isolation violations.

At the same time the packet losses due to flash congestion should be avoided, other flow requirements should also be fulfilled. Elephant flows transfer large amounts of data, then they demand high throughput but are flexible regarding delays. Hence, these flows should be organized to utilize the maximum available bandwidth of the data center. Besides, the elephant flows use all link capacity, which may cause congestions that affect mice flows. Various proposals use multiple paths provided by data center network topology to avail the links bandwidth capacity. One technique that uses multiple paths to increase overall throughput is the Equal Cost MultiPath (ECMP). The routing protocol calculates multiple minimal paths with the same cost and uses a hash function to spread flows in the different paths. Then, the flows are expected to be randomly and evenly distributed in the multiple paths. Link layer protocols already use ECMP such as Transparent Interconnection of Lots of Links [4] and 802.1aq Shortest Path Bridging [14]. Valiant Load Balancing [10] is similar to ECMP, but to select the flow path, the flow sending server randomly picks up a intermediate switch to forward the flow to.

The proposal Smart Path Assignment In Networks (SPAIN) explores the path diversity of data center topologies without any changes in the common of the shelf switches [6]. SPAIN uses an offline greedy algorithm to configure Virtual Local Area Network (VLAN) trees, so that each tree is built based on minimal path size and link reuse. The use of VLAN trees allows an easy multipath setup just by configuring different VLANs in the switches. Further, this approach requires minimal switch features, including VLAN-based Media Access Control (MAC) address learning and storage. SPAIN also runs an online algorithm in servers which test the connectivity of the destination and randomly selects a tree to send a flow.

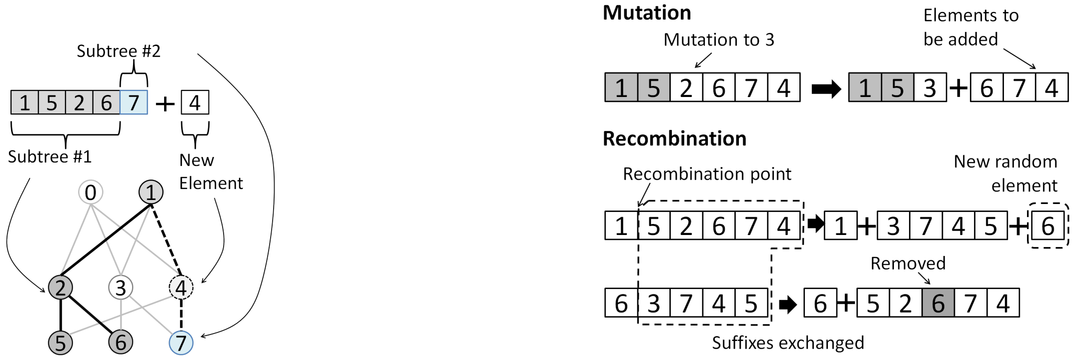
The MultiPath TCP (MPTCP) [15] divides a TCP flow into several subflows and send each in a different path, so each subflow has its own congestion control. The MPTCP approach tries to send each subflow at maximum rate available in each path to use all available bandwidth for the flow. Since this approach changes the normal TCP operation, it requires modifications in the guest operational system (OS).

The Hedera proposal detects the elephant flows and schedules them in different paths of the data center [5]. Hedera uses a NOX centralized network controller to gather information and manage OpenFlow-enabled switches. The NOX controller detects big long-live flows and periodically runs a simulated annealing algorithm to distribute these flows into different paths to maximize transmission rates. These proposals deal with the organization of the flows into paths of data center and, except MPTCP, are suited for cloud data center because they maintain tenants protocols and applications without modifications. Yet, the random path selection used in protocols such as ECMP and SPAIN, does not account for flow path collision which overloads links and degrades performance.

This paper proposes a multipathing scheme that generates optimized paths with genetic algorithm. The approach suits the cloud data center, so it does not require any tenant modification. The modifications proposed are in infrastructure provider and are maintained at a minimum level. The multipathing is accomplished in two phases: Multipath Configuration and Multipath Selection. In Multipath Configuration phase, the genetic algorithm calculates the multiple paths that can be used by the flows. Then, the multiple paths are configured in the network devices. The Multipath Configuration is an offline phase; it configures the devices when the network is not yet in operation or when topology changes occur. In the Multipath Selection phase, the flow path is chosen online.

III. Proposed Two-Phase Multipathing Scheme

The proposed multipath scheme explores path redundancy of data center topologies by using smart algorithms in two phases of multipathing: Multipath Configuration with genetic algorithm and the Multipath Selection with heuristics based on the network usage. First, the scheme generates several independent trees to interconnect the servers of the data center using genetic algorithm, in such way to minimize the path sizes and also maximize the link usage diversity. Each tree is configured in the switches with a different VLAN tag. The



(a) The genotype of an individual.

(b) Operation on the genotype.

Figure 1. Genetic algoritm for tree generation: a) Individual representation as a switch arrangement; b) Mutation and recombination operations on the genotype.

configuration is performed offline, before the operation of the data center, thereby it does not delay the configuration and scheduling of flows. A network controller can take advantage of Simple Network Management Protocol (SNMP) to gather the topology and configure the VLAN tags in switches. In each tree there is only one path between each pair of servers. The use of several trees at the same time to forward the flows spreads the traffic, which increases the aggregated throughput, load balances traffic between the links and decreases the failure impacts of burst losses due to overloaded links. One tree is selected to forward the flow, hence the flow travels by only one path which avoid packet reordering at the receptor. The tree selection use heuristics based on network usage. Before sending a flow, the virtualization server selects a tree and tags all packets of that flow. The SPAIN technique uses similar approach to explore the multiple paths of thee data center [6].

This approach requires no modification in tenants protocols, applications or operational systems. This approach also avoids isolation violations because tenants do not interact with network devices. Besides, the proposal require no modifications in network device architecture, it uses only off the shelf features such as VLAN-based MAC address learning and storage, and SNMP support. To ensure operation and reachability upon failures, all switches run Spanning Tree Protocol (STP) and use the generated tree with a default VLAN. This VLAN is used to forward packets of unknown or inactive VLANs.

Tree Generation with Genetic Algorithm: The individual of the genetic algorithm represents the switches used to interconnect the servers. The representation of an individual is an arrangement without repetition of switch identifiers (id). To generate a connected and loop-free tree, we run a procedure to add gradually switches until all Top of Rack (ToR) switches are connected, considering that servers directly connected to only one ToR switch. First, we pick a random switch and add it to a subtree. Next, we pick another random switch. If the new switch has direct connection with one of the already drawn switches, it is added to the subtree of the first directly connected switch and the link is included in the subtree. If the new switch has direct connection with switches of other subtrees, the trees are merged and the links with each subtree switch. If the new switch has no direct connection with any of the drawn switches, it is added to a new subtree. This procedure is repeated until all ToR are connected. After the inclusion of all ToR, the switches that do not connect ToR are removed from the tree. Figure 1(a) presents an example of genotype formation of an individual.

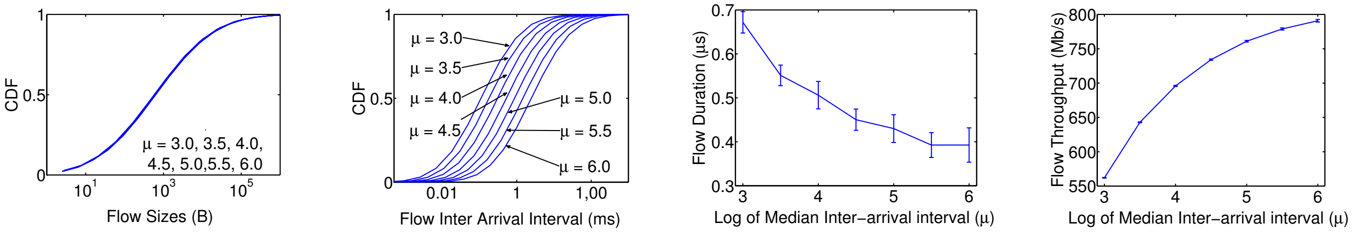
The operations on the genotype should maintain the coherence, both ToR connection and loop-free. The mutation operation selects and changes at random a switch id of the genotype, and the subsequent switches are added in order with the same procedure of tree generation. If the resulting tree does not connect all ToR switches, more random switches are appended to the genotype. The recombination operation first randomly chooses a position by which each genotype is preserved. The suffixes of both genotypes are exchanged and added to the genotype with the same inclusion procedure of mutation operation. Obviously, the duplicated switches are removed before the suffix inclusion. The operations are exemplified in Figure 1(b).

The phenotype evaluates the tree with two objective functions. The first function calculates the mean distance between ToR switches of a tree and the second calculate the sum of the number of times a link of a tree is used to compose other trees. Therefore, better phenotypes privilege smaller trees with least used links. The comparison of individuals considers better the individual whose one of its objective function is better and the other is at least equal to compose the Pareto front.

The algorithm has five steps: 1) Initialization, 2) Parent Selection, 3) Recombination, 4) Mutation, and 5) Survivor Selection. In the 1) Initialization, the algorithm is initialized with a certain number of random individuals and then it loops for a limited number of generations the steps 2) to 5). In step 2) Parent Selection, individual pairs are selected to recombine proportionally to the phenotype quality. The pairs are recombined in step 3) Recombination to double the population. Then the individuals are mutated in step 4) Mutation, and in step 5) Survivor Selection, only best half of the population survive for the next generation.

Multipath Selection Procedure: To accomplish the Multipath Selection, we define the following heuristics:

- Random selection (RND): The path is selected randomly with uniform probability;
- Least used paths selection (LUP): Each time a path is



(a) CDF of flow sizes.

(b) CDF of inter-arrival intervals.

(c) Expected flows durations.

(d) Expected flows throughputs.

Figure 2. Sanity test of the simulator of two nodes connected. 2(a) and 2(b) show the traffic models used in simulations according to measures of [11]. 2(c) and 2(d) show the resulting duration and throughput for the different μ values. Reducing μ, the logarithm of the median inter-arrival interval, more flows are active at the same time and contend for the available bandwidth, which contributes for flow rate reduction and flow duration increase.

used, the probability of choosing that path decreases;

- Least used links selection (LUL): Each time a link of a path is used, the probability of choosing that path which contains that link decreases.

It is worth mentioning that the LUP and LUL selection heuristics must access a database containing the information of paths and links usage, which may delay the path selection.

IV. Developed Discrete Event Flow Simulator

To evaluate the multipathing techniques, we developed a discrete event simulator which models data transmission in the network as data flow. The objective of the work is multipathing techniques performance evaluation, then a flow simulation model allows greater scale simulations when compared to packet simulator such as NS3[1]. The developed flow simulator creates an event queue ordered by the time it happens. At each step, an event is removed from the queue and time simulation time updated. Each event triggers other events, which are mostly arrival or departure of flows. The simulation stops when there are no events in queue or it reaches a limit time.

The flow model: A flow is defined by the tuple (source ToR switch, destination ToR switch, byte size, current transmission rate, current transmitted bytes). The link bandwidth division assumes the flows are in stationary stage. Thus, the flow transmission rate is calculated as the fair share of the most contended link by a Max-Min fairness algorithm. If a flow transmission rate has already been defined by more contended link, the remaining bandwidth are equally shared by the remaining flows. This model considers an optimistic flow model which has no acknowledgement stream, no loss, no slow start. The total bytes transferred by a flow includes extra TCP/IP and Ethernet header bytes. All packets have the maximum size allowed by Ethernet, except the last packet which send the remaining bytes.

The Simulation Parameters: The simulation can be configured with several parameters that modify different performance aspects of data center. The data center topology parameter indicates the type and size of the data center. Each type of topology presents different interconnections, so they determine the multipathing technique efficiency. The flow behavior configuration can be configured to alter the simulation workload. The flow size indicates how long the flow occupies the links of the path. The bigger the flow, the longer it takes to transmit it and the bigger the probability of more flows sharing the links bandwidth. The simulations of this paper consider a lognormal distribution to generate the flow sizes with parameters $\mu = 7$ and $\sigma = 2.8$ according to empirical measures presented by Benson *et al.* [11]. Similarly, the inter-arrival intervals also alter the simulation workload of data centers. With smaller the inter-arrival interval, more flows are active at the same time, which increases the number of flows sharing links bandwidth. The overall result is that the flow transmission rates decreases and the flow durations increase. The inter-arrival intervals are defined by a lognormal distribution with parameters $\sigma = 2$ and μ varying to decrease the workload $\mu = \{3.0, 3.5, 4.0, 4.5, 5.0, 5.5, 6.0\}$. The μ parameter is the logarithm of the median inter-arrival interval and the μ values are chosen to model the inter-arrival intervals measured in [11]. Both models used in simulation are presented in Figures 2(a) and 2(b). Another important simulation parameter is the destination set of the flows. If the destinations are uniformly distributed between all ToR switches, it is expected that all workload is uniformly distributed to the destinations, despite the random load generation. This scenario uses all available paths intensively. On the other hand, if the traffic is condensed in few destination ToR switches, the links in destination paths are highly used while other paths are not. Hence, alternative paths which use off-path links become a good choice for flows. The simulations use both all-to-all scenario, and all-to-one to represent a more realistic skewed scenario.

We performed a sanity test with a simple two-node topology. This test show the impact of the workload increase by varying the logarithm of the median inter-arrival interval μ. Figure 2 presents the results of the test. Figure 2(a) shows the CDF of the flow sizes and 2(b) shows the CDF of the flow inter-arrival interval. The reduction of the parameter logarithm of the median inter-arrival interval μ does not affect the flows sizes distribution, but the reduction of μ reduces the inter-arrival intervals of the flows. Figures 2(c) and 2(d) show the median duration and expected throughput of flows for the different μ parameter values. The reduction of μ increases the expected duration of flows as seen in Figure 2(c). Accordingly, Figure 2(d) shows the reduction of μ decreases the expected transmission rates, as expected.

V. Simulations and Results

This paper evaluates multipathing techniques according the Multipath Configuration and Multipath Selection phases. We

[1] http://www.nsnam.org/

assume that there is no link failure and it is not necessary to reconfigure the multipaths in running simulations. The topology used is fattree [9] with four-port switches, which offers four distinct paths with the same cost to the Top of Rack (ToR) switches in different pods, and two paths to ToR same pod. This topology benefits the Equal Cost MultiPath (ECMP) technique, so we consider a fair simulation scenario for comparison. The workloads are defined by lognormal distribution as described in Section IV. We gradually increase the workload by decreasing the inter-arrival interval lognormal distribution parameter μ, the logarithm of the median inter-arrival interval. The results are presented with 95% confidence interval. The simulator runs the simulations until 1000 seconds and the results are presented as expected value of the obtained measure. Next we describe some multipathing techniques and the models used in comparison with the proposed scheme.

The Spanning Tree Protocol Model: The Spanning Tree Protocol (STP) calculates a spanning between all switches, so the switches use paths that use the tree links. The Multipath configuration of this technique considers only one the path over the tree and Multipath Selection chooses the unique path available. The simulations in this paper consider a minimal spanning tree.

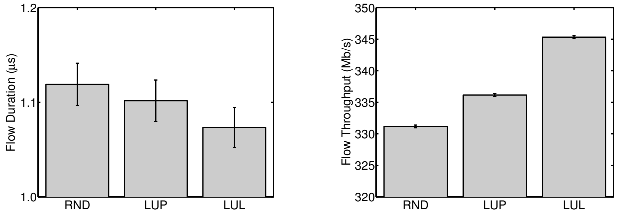
The Equal Cost MultiPath Model: The technique Equal Cost MultiPath (ECMP) considers all paths to a destination which have the same minimal cost in Multipath Configuration phase. We use a modified Dijkstra algorithm to calculate the multiple paths. In the Multipath Selection phase, the source ToR switch hashes header fields of the packets to indicate which next hop use. The simulations use `md5` hash function over source and destination ToR identifiers of the upcoming flow to select one of the available paths.

The Smart Path Assignment In Networks Model: The mechanisms created by Smart Path Assignment In Networks (SPAIN) configures several VLAN trees in switches and, during network operation, servers tag random VLAN in all flow packets. The Multipath Configuration phase calculate multiple trees offline through two algorithms, one that form sets of distinct paths and another that uses greedy algorithm to aggregate the paths between pair of servers to form the trees according to the path cost. Each time a path is used the algorithm increases the path cost. The Multipath Selection is performed by an online algorithm that runs on each server. The algorithm consults a central data base for available paths and uniformly selects one.

A. All-to-All Simulation Results

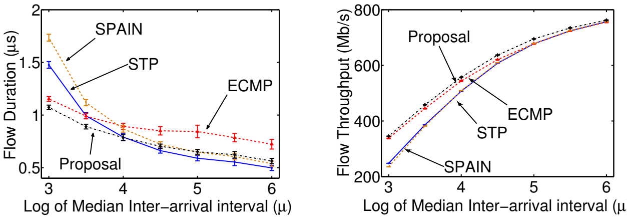
We first evaluate the multipathing techniques regarding all-to-all traffic. The flows are configured with destinations uniformly distributed between all ToR switches. The workload is expected to be uniformly distributed between paths with high link occupation.

1) Multipath Selection Heuristics Evaluation: To chose which selection heuristic the proposed scheme will use in the next simulations, we compared the following selection heuristics: random selection (*RND*), least used paths selections (*LUP*) and least used links selection (*LUL*). We used the proposed Multipath Configuration with genetic algorithm for all selection heuristics. In this simulation uses only a high workload, with lognormal($\mu = \mathbf{3.0}, \sigma = \mathbf{2}$) distribution for inter-arrival interval. Figure 3 shows the simulations results for the different heuristics. Figure 3(a) shows that the durations of the flows for all heuristics have approximately 1.1μs, with small variations. Figure 3(b) shows that in spite of the approximate duration values, the flow throughput differ from 331Mb/s to random selection (*RND*) to 345Mb/s for least used links selection (*LUL*). The LUL selection reduces the probability of using a path whose links have been already selected by several flows, as a consequence the resulting expected flow throughput is greater. The Multipath Selection heuristic used in the rest of the paper is the least used links (LUL).



(a) Expected flows durations. (b) Expected flows throughputs.

Figure 3. Multipath Selection heuristics evaluation for the proposed two-phase multipathing scheme. The heuristics compared are random selection (*RND*), least used paths selections (*LUP*) and least used links selection (*LUL*).



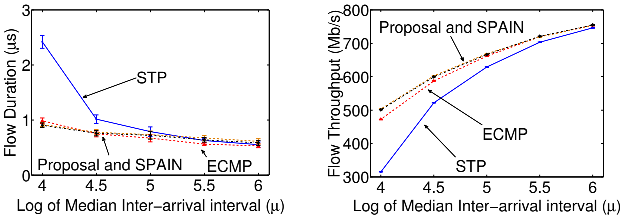
(a) Expected flows durations. (b) Expected flows throughputs.

Figure 4. Performance evaluation of the Multipathing techniques with all-to-all traffic comparing the proposed Multipath Configuration based on genetic algorithms with the techniques Spanning Tree Protocol (STP), Equal Cost MultiPath (ECMP) and Smart Path Assignment In Networks (SPAIN).

2) Multipath Configuration Evaluation: The performance of the proposed scheme is evaluated and compared with the techniques STP, ECMP and SPAIN. This simulation varies the logarithm of the median inter-arrival interval, the parameter μ, from 3.0 to 6.0 and Figure 4 shows the results. To small μ values, the expected duration of flows is smaller and the throughput greater in comparison to greater μ values, so that the techniques STP and SPAIN present worse results. Since in STP there is only one tree to forward all traffic, the tree links are always overloaded. Similarly, SPAIN creates trees through a greedy algorithm prioritizing small path sizes, so the trees may share several links. Further, the SPAIN random Multipath Selection algorithm does not account for links with high utilization, which overloads few links and degrades the performance. As the fattree topology offers four paths with the same cost, the ECMP technique successfully distributes traffic between the topology links. On the other hand, the ECMP Multipath Selection with hash functions cause collisions and many flows that should be distributed between the paths use the same path. The proposed scheme has best results with expected transmission rate gains from 7.0 to 17.6 Mb/s over ECMP.

The proposed scheme considers the link diversity in trees generation with the genetic algorithms and the path selection. Then, flow use distinct paths and links balancing the workload.

B. All-to-One Simulation Results



(a) Expected flows durations.

(b) Expected flows throughputs.

Figure 5. Performance evaluation with more realistic skewed all-to-one traffic comparing Multipathing techniques.

In the all-to-one scenario, the traffic is concentrated in one destination ToR switch. This Traffic pattern corresponds to the aggregation phase of common applications such as Map/Reduce. We expect that this scenario congests the path to the destination simulating the flash congestion of aggregation phase. Figure 5 shows the simulation results, in which the proposed scheme presents the smallest durations and the greatest throughput. The STP sole tree is in charge of all flows and then it bottlenecks all transmissions. On the other hand, the proposed scheme, SPAIN and ECMP use different paths for each flow. ECMP losses performance due to hash collisions and with $\mu = 4$ the proposed scheme has 30 Mb/s more throughput. Both proposed scheme and SPAIN consider alternative paths, including longer paths to avoid link reuse. Therefore, both have good results.

VI. Conclusion

The cloud data center networks demand new techniques to bear the hugeness of flows generated by several applications of the multiple tenants. The data center infrastructure providers should use mechanisms which do not interfere with the autonomy and isolation of the tenants. In this paper, we propose a multipathing scheme based on genetic algorithm. Our scheme allows infrastructure providers to efficiently forward the tenant flows, with no modification on tenant application or on communication protocols. We developed a flow-level simulator to evaluate and compare the proposal with other techniques from literature. The proposal distributes the traffic between links of the topology and reaches superior transmission rates in all-to-all and all-to-one traffic scenarios. The main advantage of our proposal is the online path selection heuristic which considers the least used links to choose a path for a new flow. The path configuration phase also ensures the superior performance of the proposal by calculating optimized small disjoint spanning trees. Other important feature is the fact that it only changes forwarding rules at link layer, hence it neither requires operational system nor protocol changes to be adopted. Therefore, the proposal suits multi-tenant clouds.

Our future work includes more topologies and scenarios evaluations. We also plan to extend the simulator to consider link delays and switch buffer sizes to analyze the multipath scheme impact on delay-sensitive mice flows.

Acknowledgments

The authors would like to thank CNPq, CAPES, FAPERJ, FINEP and FUNTTEL for their financial support.

References

[1] M. Bari, R. Boutaba, R. Esteves, L. Granville, M. Podlesny, M. Rabbani, Q. Zhang, and M. Zhani, “Data center network virtualization: A survey,” *Comm. Surveys Tutorials, IEEE*, vol. 15, no. 2, pp. 909–928, 2013.

[2] I. M. Moraes, D. M. Mattos, L. H. G. Ferraz, M. E. M. Campista, M. G. Rubinstein, L. H. M. Costa, M. D. de Amorim, P. B. Velloso, O. C. M. Duarte, and G. Pujolle, “FITS: A flexible virtual network testbed architecture,” *Computer Networks*, vol. 63, no. 0, pp. 221 – 237, 2014, special issue on Future Internet Testbeds - Part {II}.

[3] R. S. Couto, M. E. M. Campista, and L. H. M. K. Costa, “A reliability analysis of datacenter topologies,” in *Global Telecommunications Conference (GLOBECOM 2012), IEEE*, dec. 2012, pp. 1 –6.

[4] J. Touch and R. Perlman, “Transparent interconnection of lots of links (TRILL): Problem and applicability statement,” RFC 5556 (Informational), Internet Engineering Task Force, May 2009.

[5] M. Al-Fares, S. Radhakrishnan, B. Raghavan, N. Huang, and A. Vahdat, “Hedera: Dynamic flow scheduling for data center networks,” in *Proc. of the 7th USENIX NSDI Conf.* USENIX Association, 2010, pp. 19–19.

[6] J. Mudigonda, P. Yalagandula, M. Al-Fares, and J. C. Mogul, “SPAIN: COTS data-center Ethernet for multipathing over arbitrary topologies,” in *Proceedings of the 7th USENIX NSDI conference*, ser. NSDI’10. USENIX Association, 2010.

[7] D. Belabed, S. Secci, G. Pujolle, and D. Medhi, “Impact of ethernet multipath routing on data center network consolidations,” in *Proc. of the 4th Int. Workshop on Data Center Performance (DCPerf’14), ICDCS*. IEEE, jun. 2014.

[8] D. M. F. Mattos and O. C. M. B. Duarte, “Xenflow: Seamless migration primitive and quality of service for virtual networks,” IEEE Global Communications Conference - GLOBECOM (to appear), Dec. 2014.

[9] M. Al-Fares, A. Loukissas, and A. Vahdat, “A scalable, commodity data center network architecture,” in *Proceedings of the ACM SIGCOMM 2008 conference on Data communication*. ACM, 2008, pp. 63–74.

[10] A. Greenberg, J. R. Hamilton, N. Jain, S. Kandula, C. Kim, P. Lahiri, D. A. Maltz, P. Patel, and S. Sengupta, “VL2: A scalable and flexible data center network,” vol. 54, no. 3, pp. 95–104, Mar. 2011.

[11] T. Benson, A. Akella, and D. A. Maltz, “Network traffic characteristics of data centers in the wild,” in *Proceedings of the 10th ACM SIGCOMM Internet Measurement Conference*, 2010, pp. 267–280.

[12] M. Alizadeh, A. Greenberg, D. A. Maltz, J. Padhye, P. Patel, B. Prabhakar, S. Sengupta, and M. Sridharan, “Data center TCP (DCTCP),” in *Proceedings of the ACM SIGCOMM 2010 conference*. ACM, 2010, pp. 63–74.

[13] D. Zats, T. Das, P. Mohan, D. Borthakur, and R. Katz, “Detail: Reducing the flow completion time tail in datacenter networks,” *SIGCOMM Comput. Commun. Rev.*, vol. 42, no. 4, pp. 139–150, Aug. 2012.

[14] D. Allan, P. Ashwood-Smith, N. Bragg, J. Farkas, D. Fedyk, M. Ouellete, M. Seaman, and P. Unbehagen, “Shortest path bridging: Efficient control of larger Ethernet networks,” *Communications Magazine, IEEE*, vol. 48, no. 10, pp. 128–135, 2010.

[15] C. Raiciu, S. Barre, C. Pluntke, A. Greenhalgh, D. Wischik, and M. Handley, “Improving datacenter performance and robustness with multipath TCP,” in *Proceedings of the ACM SIGCOMM 2011 conference*, ser. SIGCOMM ’11. ACM, 2011, pp. 266–277.