International Journal For Technological Research In Engineering
Volume 5, Issue 9, May-2018 *ISSN (Online): 2347 - 4718*

OPTIMIZING TWO-AGGREGATOR TOPOLOGY USING MULTIPLE PATHS NETWORKS

T.Sathish Kumar[1], MR.P.Ramachandran[2]
Department of Computer Science, Sri Ramanujar Engineering College, Chennai-600127

Abstract: The objective of the work is to initialize data aggregation using two aggregators in a data center network, where the resource racks are permitted to come apart with their data and propel to the aggregators using multiple paths. It is to show that the problem of discovery of topology that minimizes aggregation time is NP-hard for k = 2 where k is the maximum degree of each ToR switch (number of uplinks in a top-of-rack switch) in the data center. Experimental results show that, for k = 2, our topology optimization algorithm reduces the aggregation time by as much as 73.32% and reduces total network traffic by as much as 89.5% relative to the torus heuristic, proposed in which readily proves the significant improvement in performance achieved by the proposed algorithm.
Keywords: Data center networks; aggregation time; software defined networking; big data applications; map-reduce tasks

I. INTRODUCTION

Thousands of server racks are interconnected via top-of-rack (ToR) switches to build a large data center network. Many big data applications require these racks to transfer data to a specific set of racks called aggregators in the data center. With different applications, the set of aggregators and the amount of data generated in each server rack change. Hence, using a fixed topology for the data center network may not work well for all applications. Moreover, the time required to configure the network topology to the desired one is small compared to the time required to run the application. For example, in big data applications that employ the MapReduce paradigm, the reconfiguration time is orders of magnitude lower than the time required to aggregate data from the mappers to the reducers. Software defined networking (SDN) may be used to dynamically reconfigure the data center network to improve the performance of big-data applications. Given the distribution of data on the source racks, we concentrate on the problem of constructing tree topologies that minimize the aggregation time, with the degree of each node (an aggregator or source rack) constrained by the maximum degree of a ToR switch (k, k 2) in the data center network. Please note that k is the number of uplinks in a ToR switch, current commercial ToR switches have k 10 [1]. The problem with one aggregator has already been investigated by us in [2], [3], [4]. As a first step towards generalizing the problem to multi-aggregator topologies, we focus on the problem with two aggregators in this paper. Wang et al. have observed that the aggregation time in many big-data applications is a dominant component The main contributions of this paper are TANTOS is NP-hard for k = 2.

II. RELATED WORK

This paper focuses on the two aggregator variant of the problem addressed by us in [3], [4]. In [3], [4], we have examined the single-aggregator network topology optimization with splitting (SANTOS) problem. Consequently, much of the related-work section of [3], [4] has been reproduced here and we have added in a summary of the new results obtained by us in [3], [4].
Multi-path data aggregation has been studied by many researchers in the past to seek better performance. Previous work includes research by Rao et al. [8], Xue et al. [9] among others. Rao et al. [8] have worked on providing transmission time guarantees in sending a message of finite length and obtaining a threshold on the maximum time difference between two out of order packets of a sequential message, transmitted at constant rate, from a source to a destination in a com-puter network. Xue [9] presents a polynomial time algorithm for computing an optimal multi-path end-to-end routing to transmit a given message while the previously published path-based algorithm for this problem is sub-optimal. However our problem TANTOS is markedly different from these, because TANTOS is defined on a data-center network, where the topology is constrained by the maximum degree of each ToR switch (k).

III. TANTOS IS NP-HARD WHEN k = 2

We observe that when k = 2, the aggregation tree U_1 (U_2) consists of the aggregator A_1 (A_2) as root plus at most two sub trees each of which is a chain. So, U_1 and U_2, each have at most 2 leaves. Further, each intermediate node of a sub tree chain requires 2 links while each leaf requires 1 link. Since each rack of C has to be in both U_1 and U_2, each rack of C can be assigned only 1 link in each tree and hence must be a leaf in both trees. Consequently, U_1 (U_2) can accommodate at most 2 racks from C. Hence, when jCj > 2, data aggregation is not possible using a single pair of trees U_1 and U_2. Instead, we must do some of the aggregation using one pair of trees and the rest using another. For example, we could do the aggregation for A_1 using a single tree assigning 2 links in U_1 for every rack in C and then do aggregation for A_2 using another tree U_2 in which all racks of C are assigned 2 links each. We note that some racks will use only 1 of the 2 links assigned to them. Since our TANTOS model requires the use for a single pair of trees to concurrently aggregate for both A_1 and A_2, data aggregation under the TANTOS model is infeasible when k = 2 and jCj > 2. The following theorem shows that minimizing aggregation time, under the TANTOS model, when k = 2 and jCj = 1 or 2 is NP-hard. Theorem 1. TANTOS is NP-hard when k = 2 and jCj = 1 or 2.Proof: We use the partition problem, which is known to be NP-hard. Let

s_i; 1 i n be an instance P of the partition.
From P , we create an instance T of TANTOS with k = 2, S1 = S2 = fsi g, C = fc1 = fc; cg; c2 = fc; cgg, and link bandwidth of 1 unit per sec. From the discussion preceding this theorem, it follows that, when k = 2, in every U1 and U2 for T , c1 and c2 are leaves, the sis are intermediate nodes; and no rack is split. From this, it follows that T can be aggregated in si=2 + c time iff P has a partition and that when P has no partition, the aggregation time is larger. Hence, TANTOS is NP-hard when k = 2 and jCj = 2.
To prove the theorem for the case when jCj = 1, we create the TANTOS instance T with k = 2, S1 = S2 = fsig s1,= fc1 = fs1; s1gg, and link bandwidth of 1 unit per sec. Once again, when k = 2, in every U1 and U2 for T , c1 is a leaf, all but at most one of the sis in S1 and S2 are intermediate nodes; and no rack is split. Hence, T can be aggregated in P si=2 time iff P has a partition; when P has no partition, the aggregation time is larger. Hence, TANTOS is NP-hard when k = 2 and jCj = 1. ■

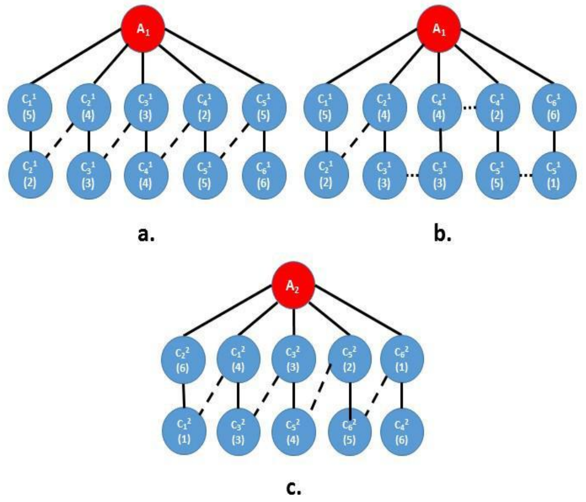

Fig. 1: The case $m_1 = m_2 = 0$; k = 2

Fig. 1 shows the aggregation tree U1 for aggregator A_1 using the depth first algorithm. In the figure, rack c^1_2, when assigned to subtree 1, takes the total data in subtree 1 to 11 units, which exceeds the OPT of 7 units. So we split c^1_2, such that it sends 2 units for data through the 1st subtree and the remaining 4 units through the 2nd subtree as shown in Fig. 2(a). This ensures that the aggregation time in U_1 is OP T.

IV. CONCLUSION

In this paper, we have explored the Two Aggregator Network Topology Optimization with Splitting (TANTOS) problem. We have proved that TANTOS is NP-hard for k = 2 using reduction from the standard 2-way Partition problem, where k is the maximum degree of a ToR switch in the data center network. We have formulated a new problem called the 3-way Partition problem and showed it to be NP-hard using reduction from the 2-way Partition problem. We have employed reduction from this newly formulated 3-way Partition problem to prove that TANTOS is NP-hard for k = 3. We have proved that TANTOS is NP-hard for k = 4 using reduction from the standard 2-way Partition problem. For k = 5 and k = 6, we have proposed polynomial time algorithms to solve TANTOS optimally by exploring all possible instances of the problem. Based on our observations in k = 5 and 6, we have conjectured that TANTOS is polynomially solvable for k > 6. Through extensive experiments, we illustrated the improved performance by our optimal algorithm for k = 6 compared to a 3D extension of the 2D torus heuristic proposed by Wang et al. in [1]. Our algorithm reduced the data aggregation time and total network traffic by up to 83:32% and 99:5%, respectively relative to Wang's heuristic.

REFERENCES

[1] G. Wang, T.S. Eugene Ng , A. Shaikh, ”Programming your network at run-time for big data applications”, Proceedings of the first workshop on Hot topics in software defined networks (HotSDN), 2012.
[2] S. Das, S. Sahni, ”Network Topology Optimization for Data Aggre-gation”, IEEE/ACM International Symposium on Cluster, Cloud, and Grid Computing (CCGrid), 2014, pp 493-501.
[3] S. Das, S. Sahni, ”Network topology optimization for data aggregation with splitting”, IEEE International Symposium on Signal Processing and Information Technology (ISSPIT), 2014, pp 398-403.
[4] S. Das, S. Sahni, ”Network topology optimisation for data aggregation using multiple paths”, International Journal on Metaheuristics, 2015, pp 115-140.
[5] S. Das, S. Sahni,”Two-aggregator network topology optimization with splitting”, IEEE Symposium on Computers and Communication (ISCC), 2015, pp 683-688.
[6] R. L. Graham, ”Bounds on Multiprocessing Timing Anomalies”, SIAM JOURNAL ON APPLIED MATHEMATICS, 1969, Volume 17, Number 2, pp 416-429.
[7] E. G. Coffman Jr., R. Sethi, ”A generalized bound on LPT sequencing”, Proceedings of ACM SIGMETRICS conference on Computer performance modeling measurement and evaluation, 1976.
[8] N. S. V. Rao, S. G. Batsell, ”QoS Routing via multiple paths using bandwidth reservation”, Proceedings of the IEEE INFOCOM, 1998, pp 11-18.
[9] G. Xue, ”Optimal multi-path end-to-end data transmission in networks”, Proceedings of ISCC, 2000, pp. 581-586.
[10] A. Hammadi, L. Mhamdi, ”A survey on architectures and energy efficiency in data center networks”, Computer Communications, 2014, vol. 40, no. 0, pp. 1 21.
[11] D. Kliazovich, P. Bouvry, Y. Audzevich, S. Khan, ”Greencloud: a packet-level simulator of energy-aware cloud computing data centers”, Global Telecommunications Conference (GLOBECOM), 2010, pp. 1-5.

[12] Y. Chen, R. Griffith, J. Liu, R.H. Katz, A.D. Joseph, ”Understanding tcp incast throughput collapse in datacenter networks”, Proceedings of the 1st ACM Workshop on Research on Enterprise Networking, WREN, 2009, pp. 73-82.

[13] Y. Zhang, N. Ansari, ”On mitigating tcp incast in data center networks”, INFOCOM, 2011, pp. 51-55.

[14] A. Phanishayee, E. Krevat, V. Vasudevan, D.G. Andersen, G.R. Ganger, G.A. Gibson, S. Seshan, ”Measurement and analysis of tcp throughput collapse in cluster-based storage systems”, Proceedings of the 6th USENIX Conference on File and Storage Technologies, FAST, 2008, pp. 12:1-12:14.

[15] Vasudevan, A. Phanishayee, H. Shah, E. Krevat, D.G. Andersen, G.R. Ganger, G.A. Gibson, B. Mueller, ”Safe and effective fine-grained tcp retransmissions for datacenter communication”, SIGCOMM Comput. Commun. 2009, Rev. 39 (4), pp 303-314,

[16] Y. Zhang, N. Ansari, ”Fair quantized congestion notification in data center networks”, IEEE Trans. Commun. 2013, pp 1-10.

[17] M. Al-Fares, A. Loukissas, A. Vahdat, ”A scalable, commodity data center network architecture”, Proceedings of the ACM SIGCOMM 2008 Conference on Data Communication, SIGCOMM, 2008, pp. 63-74.

[18] A. Greenberg, J.R. Hamilton, N. Jain, S. Kandula, C. Kim, P. Lahiri, D.A. Maltz, P. Patel, S. Sengupta, ”Vl2: a scalable and flexible data center network”, Commun. ACM, 2011, 54 (3) pp 95-104.

[19] C. Guo, G. Lu, D. Li, H. Wu, X. Zhang, Y. Shi, C. Tian, Y. Zhang,Lu, ”Bcube: a high performance, server-centric network architecture for modular data centers”, SIGCOMM Comput. Commun., 2009, Rev. 39 (4), pp 63-74.

