Load balancing mechanism and routing techniques In Data Center Networks

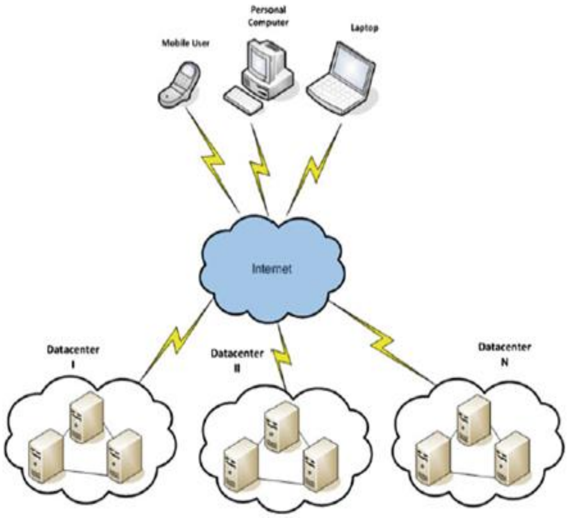
Akella Surya Srilekha
M.Tech Scholar, Department of Computer Science and System Engineering,
Andhra University College of Engineering (A), Visakhapatnam, AP, India.

Abstract: Recently, a series of data center network architectures have been proposed. The goal of these works is to interconnect a large number of servers with significant bandwidth requirements. Coupled with these new DCN structures, routing protocols play an important role in exploring the network capacities that can be potentially delivered by the topologies. This article conducts a survey on the current state of the art of DCN routing techniques. Data center networks usually employ the scale-out model to provide high bisection bandwidth for applications. A large amount of data is required to be transferred frequently between servers across multiple paths. However, traditional load balancing algorithms the article focuses on the insights behind these routing schemes and also points out the open research issues hoping to spark new interests and developments in this field. We present a comprehensive survey of recent solutions for load balancing in data center networks. First, recently proposed data center network topologies and the studies of traffic characteristics are introduced. Second, the definition of the load-balancing problem is described. Third, we analyze the differences between data center load balancing mechanisms and traditional Internet traffic scheduling. Then, we present an in-depth overview of recent data center load balancing mechanisms. Finally, we analyze the performance of these solutions and discuss future research direction. In this paper, we present a comprehensive survey of recent solutions for load balancing in data center networks. First, recently proposed data center network topologies and the studies of traffic characteristics are introduced. Second, the definition of the load-balancing problem is described. Third, we analyze the differences between data center load balancing mechanisms and traditional Internet traffic scheduling. Then we present an indepth overview of recent data center load balancing mechanisms. Finally, we analyze the performance of these solutions and discuss future research directions

***Index Terms: Wsn*, Dynamic Frame Sizing Algorithm.**

INTRODUCTION

Data Centers are the core of cloud computing as they consist of thousands of computers which are interconnected in a way to provide cloud services. A Data Center Network (DCNs) can be defined as centralized infrastructure providing several largescale computing and diversified network services like video streaming and cloud computing to its subscribed users [1].With the proliferation of internet applications, the demand for DCNs is increasing as they provide efficient platform for data storage to such applications. Figure 1 shows the block diagram of data center networks, where end users get services from data centers via the internet. In order to make datacenters quick and cost effective, dynamic resources allocation is provided by assigning Services to any server or machine in the network.



Block Diagram of Data Center Network

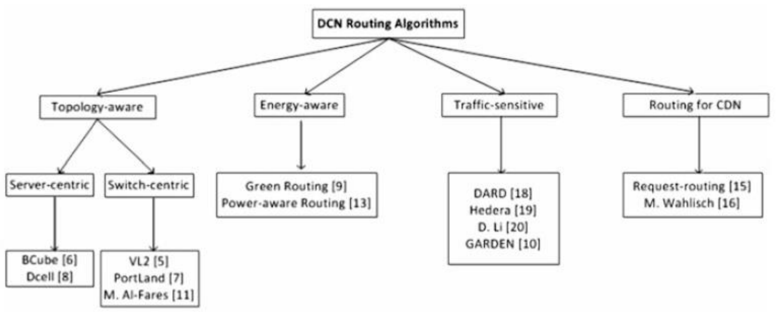
Similarly, the performance isolation between services should also be managed in DCNs. This involves lot of server to server communication and huge amount of traffic is routed among servers in a data center network. Other than the economical (cost-effective) and technical motivations behind the deployment of data center networks, they are designed to facilitate its users Data centers have some unique characteristics that make them different from other networks such as the internet or LAN. First of all, data centers are designed to deliver large scale computing services and data intensive communications such as web searching, email and network storage.

This purpose, data centers have usually high bisection bandwidth to avoid hot spots between any pair of machines in the network. Moreover, data centers have thousands of servers which are densely interconnected and normally have 1:1oversubscription ratio among the links. Since, data centers are built using commodity hardware and their failure is also a natural situation, there is a requirement of agility so that any server can service any demand. All these failure should be transparent to the client. Without agility a specific number of servers are fixed for a specific application. If number of request increase or decrease, resources becomes over loaded or underutilize respectively. Routing protocols typically determine how communications between routers take place. They determine the best routes and share network information with the irneighbors. Since, data traffic flows inside, outside, and within the servers, routing protocols are therefore, needed to route and forward data among servers in the network. Similarly, in order to explore services of data centers infrastructure, efficient routing protocols are required which are different from the traditional internet routing protocols such as OSPF and BGP. It is found that these traditional internet protocols are not suitable for DCNs due to its unique characteristics. Recently, many routing and forwarding schemes have been proposed for data centers [6–11,15,16,17–20].

For example, a simple web search request may touch 1000+ servers, and data storage and analysis applications may interactively process petabytes of data on thousands of machines. They define various traffic patterns such as one-to-one, one-to-all, and all-to-all communications. The problem known as Compact Routing (CR) consists of designing scalable routing schemes with respect to the number of network nodes. A fundamental question for DCN is how to interconnect a large number of servers with significant aggregate bandwidth requirements. To this end, many research efforts have been spent on designing scalable and efficient DCN structures. Generally, the proposed structures include server-centric structures, switch-centric structures and hybrid electrical/optical structures. In CR schemes, the memory space requirements in nodes and the time to make forwarding decisions are scalable, i.e. grow sub-linearly with respect to n, while keeping the length of any path as close as possible to the shortest one. The path stretch is defined as the ratio between the length of a path found by the scheme and the corresponding shortest path.

II. CLASSIFICATION OF ROUTING SCHEMES IN DATA CENTERS

In literature, several routing protocols have been proposed for data center networks[5–11,21–23] and they can be classified based on different criteria and approached that is used in those schemes. The criteria that we have used for classification is based on four parameters. The first parameter is topology-aware routing schemes in which we further classify the approaches based on a particular data center structure design i.e. switch-centric or server-centric approaches. The second parameter for classification is energy-aware which means that the main objective of the routing algorithm is to conserve energy or to reduce power consumption in data centers. Our third classification parameter represents the routing algorithms which are designed for either unicast, or multicast routing in data centers. All of these routing schemes in this classification have employed some type of traffic engineering approach and Hence we call it traffic-sensitive routing approaches. There are some routing schemes defined for content distribution networks (CDNs) which are data centers in essence but are given different name because of their peculiar functionality of distributing content across multiple networks. We have classified such routing schemes on ourfourth parameter i.e. routing for CDNs. The classification tree of all these protocols(which are discussed in this chapter) is shown in Fig. 2. Note that our classification based on these parameters may not be disjoint from each other and therefore, a particular traffic-sensitive routing scheme may overlap with another scheme belonging to a different category.



Classification of data centre routing schemes

Motivations and Challenges

Data centers are the foundations to support many Internet applications, enterprise operations, and scientific computations. They are large-scale and have data-intensive communications. The main challenge is how to build a scalable DCN that delivers significant aggregate bandwidth. On this question, research efforts such as BCube, DCell, PortLand, VL2, Helois, and c-Through [1–6] have been proposed in recent years. We first categorize and review the routing schemes within these structures. Then, we discuss some open questions.

Existing Solutions

Routing in Server-centric Structures. In server-centric DCNs, servers act not only as end hosts but also as relay nodes for multihop communications. Structures such as BCube [3] and DCell [2] fall into this category. To illustrate the routing, we use BCube as an example. The logic of routing in DCell and other server-centric structures is similar to that in BCube in that they all are performed by taking advantage of topological properties.

In BCube, servers are configured with multiple ports, and switches connect a constant number of servers. BCube is a recursively defined structure. A BCube0 is simply n servers connecting to an n-port switch. A BCube1 is constructed from

BCube0s and n n-port switches. In BCube, two servers are neighbors if they connect to the same switch. BCube names a server in a BCubek using an address array akak–1 … a0 (ai∈ [0, n–1], i∈ [0, k]). Two servers are neighbors if and only if their address arrays differ in one digit. More specifically, two neigh-boring servers that connect to the same level i switch are dif-ferent at the ith digit. Based on this, BCube build its routing path by "correcting" one digit at one hop from the source to destination. Figure 1 is an example of a BCube1 (n = 4) network.

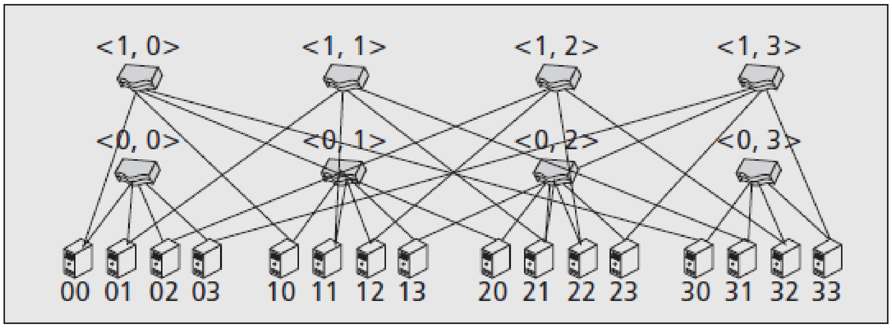


Figure 1. *An example of BCube and its address array.*

III. LITERATURE SURVEY

D. Peleg and E. Upfal,"A tradeoff between space and efficiency for routing tables,"

Two conflicting goals play a crucial role in the design of routing schemes for communication networks. A routing scheme should use paths that are as short as possible for routing messages in the network, while keeping the routing information stored in the processors' local memory as succinct as possible. The efficiency of a routing scheme is measured in terms of its stretch factor the maximum ratio between the length of a route computed by the scheme and that of a shortest path connecting the same pair of vertices. Most previous work has concentrated on finding good routing schemes (with a small fixed stretch factor) for special classes of network topologies. In this paper the problem for general networks is studied, and the entire range of possible stretch factors is examined.

M. Thorup and U. Zwick, "Compact routing schemes,"

We describe several compact routing schemes for general weighted undirected networks. Our schemes are simple and easy to implement. The routing tables stored at the nodes of the network are all very small. The headers attached to the routed messages, including the name of the destination, are extremely short. The routing decision at each node takes constant time. Yet, the stretch of these routing schemes, i.e., the worst ratio between the cost of the path on which a packet is routed and the cost of the cheapest path from source to destination, is a small constant. Our schemes achieve a near-optimal tradeoff between the size of the routing tables used and the resulting stretch

D. Coudert and G. Ducoffe, "Data center interconnection networks are not hyperbolic,"

Topologies for data center interconnection networks have been proposed in the literature through various graph classes and operations. A common trait to most existing designs is that they enhance the symmetric properties of the underlying graphs. Indeed, symmetry is a desirable property for interconnection networks because it minimizes congestion problems and it allows each entity to run the same routing protocol. However, despite sharing similarities these topologies all come with their own routing protocol. Recently, generic routing schemes have been introduced which can be implemented for any interconnection network. The performances of such universal routing schemes are intimately related to the hyperbolicity of the topology. Roughly, graph hyperbolicity is a metric parameter which measures how close is the shortest-path metric of a graph from a tree metric (the smaller the gap the better).

Y. Liu, J. K. Muppala, M. Veeraraghavan, D. Lin, and M. Hamdi,"Data center network topologies: Research proposals,"
Several new data center network topologies have been proposed by researchers, aimed at addressing many of the shortcomings of the current DCN. This chapter reviews some of the representative topologies. We classify the topologies based on their characteristics into fixed and flexible topologies. We then review their properties in terms of scale, performance and hardware redundancy.

C. Guo et al., "BCube: A high performance, server-centric network architecture for modular data centers,"
A new network architecture specifically designed for shipping-container based, modular data centers. At the core of the BCube architecture is its server-centric network structure, where servers with multiple network ports connect to multiple layers of COTS (com-modity off-the-shelf) mini-switches. Servers act as not only end hosts, but also relay nodes for each other. BCube sup-ports various bandwidth-intensive applications by speeding-up one-to-one, one-to-several, and one-to-all traffic patterns, and by providing high network capacity for all-to-all traffic. BCube exhibits graceful performance degradation as the server and/or switch failure rate increases. This property is of special importance for shipping-container data centers, since once the container is sealed and operational, it becomes very difficult to repair or replace its components

C. Guo et al.,"DCell: A scalable and fault-tolerant network structure for data centers,"
A fundamental challenge in data center networking is how to efficiently interconnect an exponentially increasing number of servers. This paper presents DCell, a novel network structure that has many desirable features for data center networking. DCell is a recursively defined structure, in which a high-level DCell is constructed from many low-level DCells and DCells at the same level are fully connected with one another. DCell scales doubly exponentially as the node degree increases. DCell is fault tolerant since it does not have single point of failure and its distributed fault-tolerant routing protocol performs near shortest-path routing even in the presence of severe link or node failures. DCell also provides higher network capacity than the traditional tree-based structure for various types of services. Furthermore, DCell can be incrementally expanded and a partial DCell provides the same appealing features. Results from theoretical analysis, simulations, and experiments show that DCell is a viable interconnection structure for data centers.

III. METHODOLOGY

Routing protocols typically determine how communications between routers take place. They determine the best routes and share network information with their neighbors. Since, data traffic flows inside, outside, and within the servers, routing protocols are therefore, needed to route and forward data among servers in the network. Similarly, in order to explore services of data centers infrastructure, efficient routing protocols are required which are different from the traditional internet routing protocols

Adaptive Routing scheme

The idea of adaptive routing is to dynamically find a best path for each flow based on the present network traffic status. Efficient adaptive routing can minimize flow collision that occurs when many flows compete for bandwidth. Adaptive routing is also inherent fault-tolerant to already happened faults: all failed nodes/links are automatically detected/avoided by the routing mechanisms; hence adaptive routing can "smooth out" any unevenness in the network to load balance the traffic to the set of available forwarding switches/links. For each flow, an adaptive path selection process should be performed first: additional start up delay may be added. Fortunately, such delay can be avoided [9]: when a source is performing path selection for a flow, it does not hold packets. The source uses a default path selecting (e.g. a randomized one) for that flow first. After the routing process completes and a better path is selected, the sources witch the flow to the new path. However, these merits do not come at no cost. The design and implementation of adaptive routing mechanism, centralized path scheduling or distributed path probing, should be provided. The next two parts present our considerations.

Centralized Path Scheduling

Centralized path scheduling requires a global view of the network status, including links/ports utilization and nodes health. We expect that a dedicated Fabric Master system periodically receives link utilization reports from switches that take part in forwarding. Routing Agents, either on the application servers or edge switches, would implement the adaptive routing mechanism for flows. We also expect a centralized Routing Scheduler: a Routing Agent sends routing requests for flows to Routing Scheduler; Routing Scheduler replies with the best path back to the sender in a timely manner.

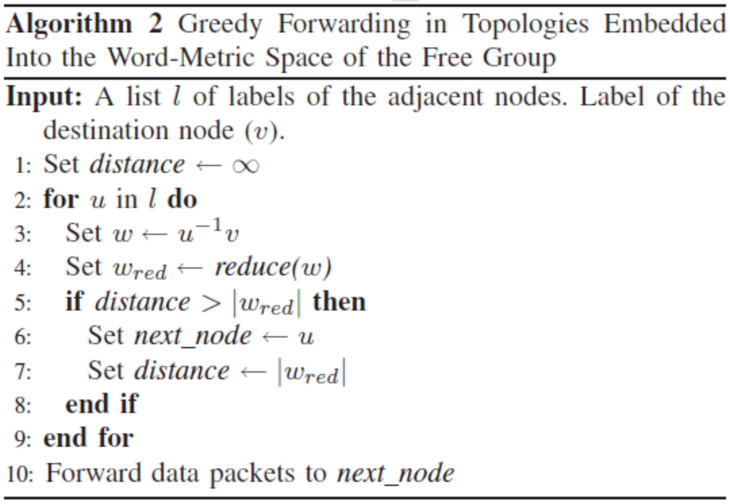
Distributed Path Probing

The concept of distributed path probing when a new flow came, the source sends probe packets over multiple paths; the passing switches process the probe packets to fill the minimum available bandwidth of their incoming and outgoing links; the destination returns probe responses to the source. When the source receives the probe responses, it selects the best path from them. For simplicity, here we use Broadest Fit (BF) algorithm to select one from available paths: flow is assigned to the broadest path that can provide the maximum bandwidth.

Greedy Forwarding Algorithms

The procedure for forwarding data packets between two nodes starts at the source node by computing the distances from its adjacent nodes to the destination node. Then, data packets are forwarded to the nearest node to the destination node. The same procedure is repeated in the previously selected node until data packets reach the destination node. The forwarding process in topologies embedded into the WMS of the FG is performed by using Algorithm 2. Note that packets can be forwarded through links that are not in the spanning tree, because nodes find among all their adjacent nodes the nearest one to the destination. For example, in Fig. 5, when routing packets from node cb to node a, node cb computes the distance in the WMS from ab to a, which is 1, and from c to a, which is 2. Then packets are forwarded to ab, although edge (cb,ab) is not in the spanning tree. The forwarding process in topologies embedded into the WMS of their underlying CG is performed by using Algorithm 3, which improves the temporal complexity with respect to Algorithm 2 and the forwarding technique proposed in [15] and [34]. Let u and v be node labels of a source and a destination nodes. If the word w is the canonical form of u−1v, then w represents the sequence of edges in the shortest path between the source and the destination node. Thus, we can select the nearest node to a destination node as the one that is connected by the edge (generator) given by the first letter in w.

This procedure works because, as we prove in our network embedding is isometric for topologies based on trees or embedded into its own CG, i.e. words in canonical form are congruent with shortest paths in the network. The core of algorithms 2 and 3 is the reduce procedure, which computes the canonical form of any word by solving the MWP.



Algorithm 2 Greedy Forwarding in Topologies Embedded Into the Word-Metric Space of the Free Group

```
Input: A list $l$ of labels of the adjacent nodes. Label of the destination node ($v$).
Set distance ← $\infty$
for $u$ in $l$ do
    Set $w \leftarrow u^{-1}v$
    Set $w_{red} \leftarrow$ reduce($w$)
    if distance > $|w_{red}|$ then
        Set next_node ← $u$
        Set distance ← $|w_{red}|$
    end if
end for
Forward data packets to next_node
```

CONCLUSION

This paper addresses load balancing, which is becoming an important research area that enables reducing completion time of flows, maximizing bandwidth utilization, and saving energy for data center networks. First, we presented data center network topologies, traffic characteristics, and main properties and objectives for load balancing mechanisms. Then, the differences between load balancing in data centers and traffic engineering in WAN were presented. Next, recent proposals of load balancing were surveyed and analyzed. We broadly classified the proposals into those that use a centralized controller to balance load and those that run on distributed network devices. As large energy consumption is a big issue in data centers, we also summarized energy-aware load balancing schemes in data center networks. We observed that centralized schemes can accurately collect global congestion information and balance load near optimally. However, centralized schemes usually have large control overheads. To reduce the overheads, 0other schemes may jointly leverage centralized and distributed approaches or only use distributed protocols. However, the schemes usually require complex infrastructure to support. Finally, we discussed design goals, design considerations and future directions with the development trend of data center networks. We foresee that more research is required in designing load balancing schemes. Especially, designing load balancing schemes for cluster computing applications, multitenant data centers and leveraging machine learning algorithms will become promising research directions. In summary, there are still a number of research issues and challenges in load balancing. Nevertheless, it is promising to address these challenges by leveraging new technologies, such as P4, machine learning. This article attempts to briefly explore current load balancing technologies and discuss future research directions.

REFERENCES

[1] D. Peleg and E. Upfal, “A tradeoff between space and efficiencyfor routing tables,” in Proc. 20th Annu. ACM Symp.Theory Comput.(STOC), New York, NY, USA, 1988, pp. 43–52. [Online]. Available:http://doi.acm.org/10.1145/62212.62217

[2] M. Thorup and U. Zwick, “Compact routing schemes,” in Proc. 20thAnnu.ACM Symp. Parallel Algorithms Archit. (SPAA), New York,NY, USA, 2001, pp. 1–10. [Online]. Available: http://doi.acm.org/10.1145/378580.378581

[3] D. Coudert and G. Ducoffe, “Data center interconnection networks arenot hyperbolic,” Theor. Comput. Sci., vol. 639, pp. 72–90, Aug. 2016.[Online]. Available: http://www.sciencedirect.com/science/article/pii/S030439751630161X

[4] Y. Liu, J. K. Muppala, M. Veeraraghavan, D. Lin, and M. Hamdi,“Data center network topologies: Research proposals,” in Data CenterNetworks (SpringerBriefs in Computer Science). Cham, Switzerland:Springer, 2013.

[5] C. Guo et al., “BCube: A high performance, server-centric networkarchitecture for modular data centers,” in Proc. ACM SIGCOMM Conf.Data Commun. (SIGCOMM), New York, NY, USA, 2009, pp. 63–74.

[6] C. Guo et al., “DCell: A scalable and fault-tolerant network structurefor data centers,” in Proc. ACM SIGCOMM Conf. Data Commun. (SIGCOMM),New York, NY, USA, 2008, pp. 75–86. [Online]. Available:http://doi.acm.org/10.1145/1402958.1402968

[7] M. Al-Fares, A. Loukissas, and A. Vahdat, “A scalable, commoditydata center network architecture,” ACM SIGCOMM Comput. Commun.Rev., vol. 38, no. 4, pp. 63–74, 2008.[Online]. Available: http://doi.acm.org/10.1145/1402946.1402967

[8] G. D. Stamoulis and J. N. Tsitsiklis, “The efficiency of greedy routingin hypercubes and butterflies,” in Proc. 3rd Annu. ACM Symp. ParallelAlgorithms Archit. (SPAA), New York, NY, USA, 1991, pp. 248–259.[Online]. Available: http://doi.acm.org/10.1145/113379.113402

[9] S. B. Akers, B. Krishnamurthy, and D. Harel, “The star graph: Anattractive alternative to the n-cube,” in Proc. Int. Conf. Parallel Process.,1987, pp. 393–400.

[10] S. B. Akers and B. Krishnamurthy, “A group-theoretic model forsymmetric interconnection networks,” IEEE Trans. Comput., vol. 38,no. 4, pp. 555–566, Apr. 1989.

[11] J. Y. Yen, “Finding the K shortest loopless paths in a network,” Manage.Sci., vol. 17, no. 11, pp. 712–716, 1971.

[12] A. Singla, C.-Y. Hong, L. Popa, and P. B. Godfrey, “Jellyfish: Networkingdata centers, randomly,” in Proc. 3rd USENIX Conf. Hot TopicsCloud Comput. (HotCloud), Berkeley, CA, USA, 2011, p. 12. [Online].Available: http://dl.acm.org/citation.cfm?id=2170444.2170456

[13] A. Valadarsky, M. Dinitz, and M. Schapira, “Xpander: Unveiling thesecrets of high-performance datacenters,” in Proc. 14th ACM WorkshopHot Topics Netw. (HotNets-XIV), New York, NY, USA, 2015,pp. 16:1–16:7. [Online]. Available: http://doi.acm.org/10.1145/2834050.2834059

[14] M. Besta and T. Hoefler, “Slim fly: A cost effective low-diameternetwork topology,” in Proc. Int. Conf. High Perform. Comput.,Netw.,Storage Anal. (SC), Piscataway, NJ, USA, Nov. 2014, pp. 348–359.[Online]. Available: https://doi.org/10.1109/SC.2014.34

[15] M. Camelo, D. Papadimitriou, L. Fàbrega, and P. Vilà, “Geometricrouting with word-metric spaces,” IEEE Commun.Lett., vol. 18, no. 12,pp. 2125–2128, Dec. 2014.

[16] S. T. Schibell and R. M. Stafford, “Processor interconnection networksfrom cayley graphs,” Discrete Appl. Math., vol. 40, no. 3, pp. 333–357,1992. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0166218X9290005U

[17] K. W. Tang and B. W. Arden, “Vertex-transitivity and routing forCayley graphs in GCR representations,” in Proc. ACM/SIGAPP Symp.Appl. Comput., Technol. Challenges (SAC), New York, NY, USA, 1992,pp. 1180–1187.

[18] M. Miller and J. Širáˇ n, “Moore graphs and beyond: A survey ofthe degree/diameter problem,” Electron. J. Combinat., vol. 20, no. 2,May 2013, Art. no. DS14.

[19] C. H. Papadimitriou and D. Ratajczak, “On a conjecture related togeometric routing,” Theor. Comput. Sci., vol. 344, no. 1, pp. 3–14,Nov. 2005. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0304397505003725

[20] R. Dhandapani, “Greedy drawings of triangulations,” in Proc. 19thAnnu. ACM-SIAM Symp. Discrete Algorithms (SODA), Philadelphia,PA, USA, 2008, pp. 102–111. [Online]. Available: http://dl.acm.org/citation.cfm?id=1347082.1347094

[21] R. Kleinberg, “Geographic routing using hyperbolic space,” in Proc.26th IEEE Int. Conf. Comput. Commun. (INFOCOM), May 2007,pp. 1902–1909.

[22] D. Eppstein and M. T. Goodrich, “Succinct greedy geometric routingusing hyperbolic geometry,” IEEE Trans. Comput., vol. 60, no. 11,pp. 1571–1580, Nov. 2011.

[23] J. Newsome and D. Song, “GEM: Graph EMbedding for routing anddata-centric storage in sensor networks without geographic information,”in Proc. 1st Int. Conf. Embedded Netw. Sensor Syst. (SenSys), New York,NY, USA, 2003, pp. 76–88.

[24] M. R. Bridson and A. Haefliger, “Metric spaces of non-positivecurvature,” in Grundlehren der MathematischenWissenschaften.Berlin, Germany: Springer, 1999. [Online]. Available: http://opac.inria.fr/record=b1095970

[25] D. J. S. Robinson, “A course in the theory of groups,” in Graduate Textsin Mathematics, vol. 80. Berlin, Germany: Springer-Verlag, 1996.

[26] C. H. Papadimitriou and D. Ratajczak, “On a conjecture related togeometric routing,” in Proc. ALGOSENSORS, 2004, pp. 9–17.

[27] D. B. A. Epstein, Word Processing in Groups. Natick, MA, USA:A. K. Peters, 1992.

[28] J. Meier, Groups, Graphs Trees. Cambridge, U.K.: CambridgeUniv. Press, 2008. [Online]. Available: http://dx.doi.org/10.1017/CBO9781139167505

[29] S. Even and O. Goldreich, “The minimum-length generator sequenceproblem is NP-hard,” J. Algorithms, vol. 2, no. 3, pp. 311–313, 1981.

[30] C. Boulinier, A. K. Datta, L. L. Larmore, and F. Petit, “Space efficientand time optimal distributed BFS tree construction,” Inf. Process.Lett., vol. 108, no. 5, pp. 273–278, Nov. 2008. [Online]. Available:http://dx.doi.org/10.1016/j.ipl.2008.05.016

[31] D. B. A. Epstein, A. R. Iano-Fletcher, and U. Zwick, “Growthfunctions and automatic groups,” Experiment. Math., vol. 5, no. 4,pp. 297–315, 1996.[Online]. Available: http://projecteuclid.org/euclid.em/1047565448

[32] D. Knuth and P. Bendix, “Simple word problems in Universal Algebras,”in Automation of Reasoning (Symbolic Computation), J. Siekmann andG. Wrightson, Eds. Berlin, Germany: Springer, 1983, pp. 342–376.

[33] M. Camelo, P. Vilà, L. Fàbrega, and D. Papadimitriou, “Cayley-graphbaseddata centers and space requirements of a routing scheme usingautomata,” in Proc. IEEE 34th Int. Conf. Distrib.Comput. Syst. Workshops(ICDCSW), Jun. 2014, pp. 63–69.

[34] M. Camelo, D. Papadimitriou, L. Fàbrega, and P. Vilà, “Efficient routingin data center with underlying cayley graph,” in Proc. 5th WorkshopComplex Netw.CompleNet, 2014, pp. 189–197.

[35] D. F. Holt, B. Eick, and E. A. O’Brien, Handbook of ComputationalGroup Theory (Discrete Mathematics and its Applications).Boca Raton, FL, USA: Chapman & Hall, 2005. [Online]. Available:http://opac.inria.fr/record=b1102239

[36] D. Holt. (2009). KBMAG Package: A Knuth-Bendix on Monoids,and Automatic Groups. Accessed: Jan. 15, 2015.[Online]. Available:http://homepages.warwick.ac.uk/mareg/kbmag/

[37] M. Gromov, “Hyperbolic groups,” in Essays in Group Theory (MathematicalSciences Research Institute Publications), vol. 8. New York,NY, USA: Springer, 1987, pp. 263–275.