JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 36, 1093-1105 (2020)
DOI: 10.6688/JISE.202009_36(5).0011

Efficiently Controlling Wireless Data Centers with Exact Topology Adjustment Strategy

YI ZHANG AND TENGFEI CAO
State Key Laboratory of Networking and Switching Technology
Beijing University of Posts and Telecommunications
Beijing, 100876 P.R. China
E-mail: piliyi@126.com

As wireless transmission technology advances, wireless data centers (WDCs) are more and more widely used as computing infrastructures benefiting from its flexibility and low complexity of cabling. Software-defined networking (SDN) brings new convenience in topology design of wireless data center networks (DCNs) which has a great effect on the performance of data centers. In this paper, we introduce the controllability as a new consideration for wireless data center designers. We defined a new measure for controllability evaluation and compared the existing WDC architectures, uncovered their diversity on properties of dynamics. Then we propose a novel exact topology adjustment strategy to control WDCs more efficiently. Mathematical reasoning and experimental results show the correctness and effectiveness of our method. Our method can not only significantly improve the controllability of existing WDCs, but also applicable in accurate architecture design of wireless network systems.

Keywords: wireless data centers, software-defined networking, exact controllability, network topology design, topology adjustment

1. INTRODUCTION

In recent years, we are undergoing a tremendous growth in mobile client applications, which spawns much more diversify demand for the performance of data centers. Along with the number of hosted servers continues to grow exponentially, the scalability, for example, has become a key bottleneck in the capacity improvement of data centers. Many researchers are committing themselves to the DCN architecture design to ease the unprecedented demands of extensive use of cloud-based services and massive data transmission in data centers [1, 2]. The fixed topology and cabling layout of wired DCNs seriously restricts the network performance improvement because modifying deployed wired networks can be much costly and complex. In order to further provide more flexible network configuration, software-defined networking technology has been adopted and deployed by many enterprises. Researchers begin to investigate wireless communication to solve cabling complexity problems in DCNs. Internet companies like Microsoft

Received September 16, 2019; revised October 25, 2019; accepted December 6, 2019.
Communicated by Xiaohong Jiang.

and Google recently attempt to adopt 60 GHz wireless links into wired data centers [3]. DCNs augmented by wireless links emerge in such situations and have been concerned by many researchers [3, 4]. Wireless connection makes the WDC topology more flexible, but meanwhile makes the wireless architectures design a more important issue to enhance the performance. Most WDCs are implemented by adopting 60 GHz RF technology in conventional wired DCNs to emulate well-known topologies. Directional antennas with narrow beams are usually mounted on top-of-rack switches and used for intra-rack communication with servers by wireless channel.

As the threat of DDoS attacks sourcing from IoT terminals on WDCs becomes more and more serious, the resistance of DCNs to cyberattacks[5, 6], and the capacity of recovering from cascading failure and some other destructions after been attacked was also generally concerned in DCN architecture design [7]. So the controllability of WDCs has become much more important than ever before [8, 9]. Meanwhile, the concepts of Green IT, especially green networking technology, generates new demands on precise management of the network equipment to improve energy efficiency and reduce network consumption in WDCs [10, 11]. With the purpose of defending against DDoS attacks, recovering the paralyzed network links from cascading failure, or shutting down redundant servers and routers in green datacenters, WDC managers always need to exactly control some of the WDC nodes to inject control signals on condition that control functional components have been deployed on such nodes beforehand. Otherwise, WDC managers have to finish the reconfiguration manually device by device, which is quite time-consuming. It is obvious unfeasible to deploy control functional components on every node in a WDC due to the very high expenditure in hardware and software overhead. One natural and core questions is how to select such a set of driver nodes to deploy control components, which can both achieve the full control of the entire WDC network and minimize the total overhead?

Before answering this question, one may wonder that (1) do the existing WDCs perform well in the viewpoint of controllability? (2) If not, how to improve their controllability by minimum costs? Actually, the congestion control problem in DCNs has been widely investigated by many researchers [12, 13]. However, these studies mainly focus on flow control of network traffic, and are difficult to understand the behavior of dynamical processes occurring on DCNs. From the perspective of the dynamics of complex networks, the controllability of a WDC network is closely linked to its topology design. So Topology adjustment is one of the most straightforward method to minimize the set of driver nodes, thus reduce the deployment cost of control functional components. So far as we know, there is no such work which has fully investigated the control issues of wireless DCNs. The motivation of this paper is to bridge this gap and answer the aforementioned questions using a novel integrated methodology. Key contributions of this work are in three aspects: (1) We reevaluate the performance of existing DCN architectures in the viewpoint of controllability; (2) We propose a feasible method to improve the controllability performance of existing WDCs by adjusting their topology with very low cost; (3) We prove the correctness of our method by rigorous mathematical reasoning, and verified the effectiveness by practical experiments.

The remainder of this paper is organized as follows: in Section 2, we introduce some preliminaries on controllability theory of complex networks. The controllability of some existing WDC architectures are evaluated and compared in Section 3. In Section

4, a feasible method is expatiated to improve the controllability of WDC architectures by making small changes on physical topology. Section 5 is our future work and conclusion.

2. BASIC CONTROL THEORY OF COMPLEX NETWORKS

As one of the most challenging problems in modern network science, controlling complex networks has been studied for decades, but it still remains an open problem. Until the structural controllability was proposed by Liu *et al*. in 2011 [14], the passion of researchers was ignited again. Many insightful results were revealed by following their thoughts [15], the controllability theory of complex networks has been greatly enriched. However, these results are only applicable to directed networks, bi-directed ones such as WDC networks can't be well analyzed. A ground-breaking contribution to this issue was made by Yuan *et al*. [16], a universal tool for exactly exploring the controllability of complex networks with arbitrary structures and configurations of link weights was proposed. Our controllability evaluation is based on this contribution. We consider a WDC with N nodes, governed by the following linear dynamics [16]:

$$\dot{x} = Ax + Bu \tag{1}$$

where the vector $x = (x_1, x_2, \cdots, x_N)^T$ is the states of all nodes, and the matrix $A \in \mathbf{R}^{N\times N}$ is the coupling matrix of the network, *i.e*. the adjacency matrix of DCN. Since wireless DCN is usually a bi-direction network, A is an unweighted real symmetric matrix. B is the $N \times m$ control matrix and u is the input control signal of m controllers. In this paper, we consider the servers and switches as the same nodes in DCNs. According to the PBH rank condition, the DNC described by Eq. (1) is fully controllable only when

$$rank(cI_N - A, B) = N \tag{2}$$

where c is an arbitrary plural and I_N is the identity matrix of dimension N. Based on this condition, Yuan *et al*. further proved that for any network with adjacency matrix A, the minimal number of driver nodes, denoted by N_D, is determined by the maximum geometric multiplicity $\mu(\lambda_i)$ of the eigenvalue λ_i of A:

$$N_D = \max_i \mu(\lambda_i) \tag{3}$$

where $\lambda_i,\ i = 1, 2, \cdots, k$ is a nonidentical eigenvalue in matrix A, and $\mu(\lambda_i) = N - rank(\lambda_i I_N - A)$ [16]. Furthermore, since A is symmetric, the geometric multiplicity of eigenvalue λ_i equals to the algebraic multiplicity, $\delta(\lambda_i)$, of λ_i. Thus, N_D can be determined by:

$$N_D = \max_i \delta(\lambda_i). \tag{4}$$

We denote the eigenvalue with maximum geometric multiplicity by λ_M. A corresponding set of control nodes can be obtained by crossing off a maximal linearly independent group from the column vectors of $\lambda_M I_N - A$. We will discuss it later.

It is worth noting that the exact controllability is quite different from structural controllability. For the later one, adding more links to a network will make it more controllable, *i.e*. controllable by less driver nodes. But for the former, too few links or too many links will increase the minimum number of driver nodes, which makes the controllability more complicated.

3. CONTROLLABILITY ANALYSIS OF WDC ARCHITECTURES

Before evaluating the controllability of mainstream WDCs, we first need to define metrics for controllability. Since the server is a crucial component of WDCs that requires precise control, we define the controllability as the ratio of the number of servers N_S in the WDC architecture to N_D:

$$n_D = \frac{N_S}{N_D}. \tag{5}$$

For ease of comparison, the WDCs hereinafter are assumed to be consist of switches with maximal 4 links (no matter wired or wireless) and servers with wireless transceivers. In current WDC architectures, only the switches which directly connect to servers has wireless transceivers attached, such as top-of-rack (ToR) switches, while the links between switches are still wired. We follow this setting in our research, and take the mainstream WDCs which emulating Three-tier, FatTree [17], Clos [18], Bcube$_1$ [19], Dcell$_1$ [20] and FiConn$_1$[21] architectures as examples. For the recursive architectures, such as Bcube$_1$, Dcell$_1$, FiConn$_1$, we still denote them as Bcube, Dcell and FiConn hereinafter for short without special notification. We proceed our analysis by the following steps:

1. Formulating the coupling matrix. For the tree/Clos based architectures, we serially number the nodes from top to bottom, left to right, and from outside to inside anticlockwise for recursive ones. Then formulate the adjacency matrix A. Note that the number of nodes is irrelevant to the result.
2. Using the SVD method to compute the eigenvalues of matrix A, then select the eigenvalue λ_M with maximum algebraic multiplicity.
3. Calculating the controllability n_D by Eq. (5).
4. Comparing the controllability of different WDC architectures.

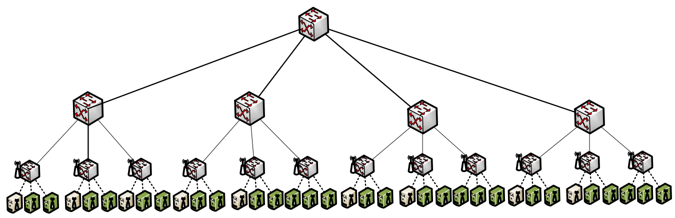


Fig. 1. A WDC of Three-Tier architecture with driver nodes painted green.

We reprint the architectures with the driver nodes selection results in Figs. 1-6.
The exact controllability comparison of the aforementioned WDC architectures are specified in Table 1. Since FatTree can be considered as a folded Clos network, they shared the same level of controllability, and both are with the worst performance in controllability.

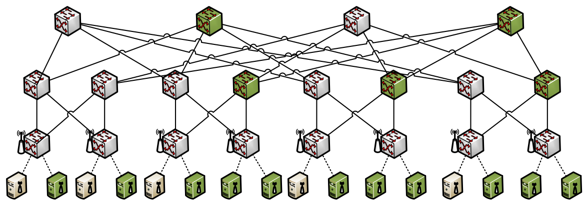


Fig. 2. A WDC of FatTree architecture with driver nodes painted green.

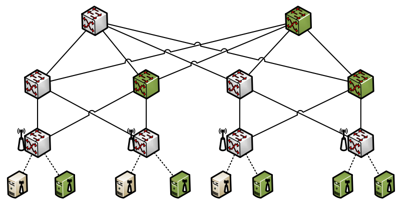


Fig. 3. A WDC of Clos architecture with driver nodes painted green.

Whereas 3-Tier topology supports more servers at the same cost compared to FatTree and Clos, the controllability is slightly better (1.33 vs 1). As hierarchical recursive architecture, Bcube is significantly superior in controllability with $n_D = 1.6$. Dcell and FiConn is substantially different. Dcell has a much better performance with $n_D = 3.33$ while FiConn is even better and attains a very good controllability with $n_D = 4$. The really interesting question is: what plays the decisive role to controllability of WDCs? It is quite notable that the λ_M of Dcell is -1 while the others are all 0. According to the exact controllability theory, an obvious explanation for this difference is that WDCs with $\lambda_M = 0$ are sparse while Dcell is dense. This question will be further discussed in next section.

Now we have determined the minimum number of driver nodes for aforementioned WDC architectures, in order to implement exact control of them, we need to select a set of such driver nodes to inject control signal. Review the exact controllability theory, the corresponding set of control nodes can be obtained by crossing off a maximal linearly independent group from the column vectors of $\lambda_M I_N - A$. It suffices to implement elementary row transformations on the matrix $\lambda_M I_N - A$ to find a maximal linearly independent group on the column vectors. Then we cross off the corresponding serial numbers from the column number set, the remaining serial numbers constitute the index set of driver nodes. We calculate a series of instances of driver nodes and paint them green in Figs. 1-6.

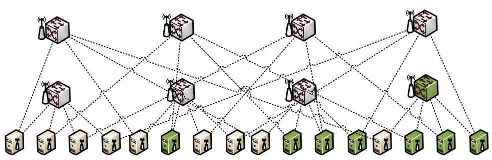


Fig. 4. A WDC of Bcube architecture with driver nodes painted green.

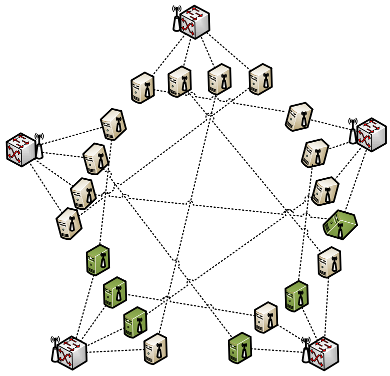


Fig. 5. A WDC of Dcell architecture with driver nodes painted green.

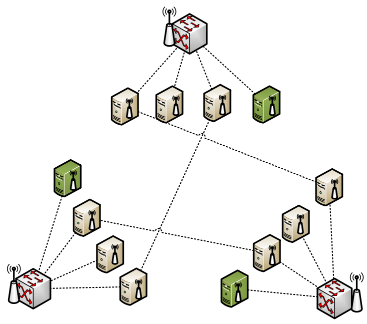


Fig. 6. A WDC of FiConn architecture with driver nodes painted green.

4. EXACT CONTROL OF WDC ARCHITECTURES

By comparison we find that Dcell and FiConn are superior to other WDC architectures in controllability. Evaluating and ranking are not enough, our ultimate goal is to improve the controllability systematically in WDC architecture design. We come back to investigate the decisive factor of controllability. In structural controllability theory, the denser a network is, the fewer driver nodes are needed to control it [14]. Both adding new links and decreasing the degree heterogeneity can improve the structural controllability of a given network. Unfortunately, this does not hold in exact controllability theory. For instance, the star and fully connected networks with N nodes are two extreme cases in degree heterogeneity. The star network is the hardest to be structurally controlled, and for fully connected networks, 1 driver node is quite enough. While in order to implement exact control, $N-2$ and $N-1$ control nodes are needed respectively. It seems that too many and too few links both makes exact control difficult. In essence, it is the coupling degree of adjacency matrix that determined the controllability. Therefore, we mainly focus on implementing few variations on the adjacency matrix A (*i.e.* the topology of WDC architecture) to expand the maximal linearly independent group of the column vectors of the matrix $\lambda_M I_N - A$ to get better performance on controllability.

Table 1. The controllability comparison of WDC architectures.

DCNs	Num. of Servers N_S	λ_M	Num. of driver nodes N_D	Controllability n_D
Tree	36	0	27	1.33
FatTree	16	0	16	1
Clos	8	0	8	1
Bcube	16	0	10	1.6
Dcell	20	-1	6	3.33
FiConn	12	0	3	4

4.1 Control Efficiency Enhancement Using exact Topology Adjustment Strategy

We have known that the controllability of WDC architecture with adjacent matrix A is fully determined by the rank of $B = \lambda_M I_N + A$, *i.e.* the maximal linearly independent group on its column vectors. By the analysis of current WDCs above, we note that λ_M is either 0 or -1. This is not a coincidence. For a sparse network, the maximum geometric multiplicity occurs at the eigenvalue $\lambda_M = 0$ with high probability[16]. Thus we have $B = A$. This observation motivates us to expand the maximal linearly independent group of A's column vectors instead of B, which can be accomplished by simply adjusting a few elements of A, corresponding to adding or cutting off links in the original topology.

We take Dcell as an example to expatiate how to improve controllability by adding new links or cutting off redundancy links. The adjacency matrix of Dcell is of 25×25 dimensional, and the λ_M is -1. We need to analyze the matrix $-\lambda_M I_{25} - A$, or equivalently, $\lambda_M I_{25} + A$. We number the nodes of Dcell as shown in Fig. 7.

The initial driver nodes are numbered 17, 19, 21, 23, 24, 25. We implement elementary row transformations on the column vectors to get the row canonical form of $\lambda_M I_{25} + A$. It can be easily obtained that the column vectors are the first 18 unit basis vectors, $(1, 0, \cdots, 0)^T$, $(0, 1, 0, \cdots, 0)^T$, $\cdots$, $(0, \cdots, 0, 1, 0, 0, 0, 0, 0)^T$, except the driver columns. We keep the link relationships of the non-driver nodes unchanged, and change the link relationship of 17th node, make it to be one of the 19th-25th basis vector.

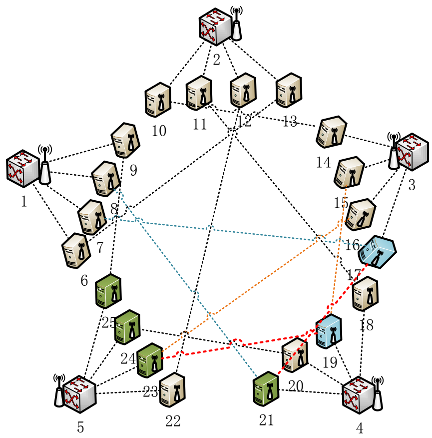


Fig. 7. Improved WDC topology of Dcell architecture.

We link it to the 21st node after cutting off the links $7 \leftrightarrow 17$ and $8 \leftrightarrow 21$ (illustrated by dashed blue lines in Fig. 7), then add a new link to $17 \leftrightarrow 21$(illustrated by dashed red line in Fig. 7). We denote the adjacency matrix of the improved network by A_1. Recalculation of controllability shows that N_D of A_1 is turned to 5, the 17th node is not driver node any more, its n_D has been improved to 4, and $\lambda_M^{(1)}$ is still -1. We continue to carry on this procedure, recalculate the $\lambda_M^{(1)} I_{25} + A_1$ and disconnect the links $19 \leftrightarrow 15$ and $23 \leftrightarrow 16$ (illustrated by dashed orange lines in Fig. 7), and then add a new link $19 \leftrightarrow 23$(illustrated by dashed red line in Fig. 7). After this iteration, we get adjacency matrix A_2 with $N_D = 4$, $n_D = 5$ and $\lambda_M^{(2)} = -1$. The driver nodes become $21,\ 23,\ 24,\ 25$. We need only 4 driver nodes to exactly control the improved Dcell architecture. The released driver nodes 17 and 19 are repainted blue in Fig. 7. This adjustment procedure can be continued until no more links are permitted to add. Now that it is unreasonable to link any two links connected to the same mini-switch, we stop here for the adjustment. The final topology of improved Dcell is illustrated in Fig. 7.

To formulate our adjustment method, we give a general algorithm to controllability enhancement described in Algorithm 1. We complement some extra explanations on the

Algorithm 1 : The Exact Topology Adjustment to Control WDC Architectures.

Input: The adjacency matrix A with dimension $N \times N$; The threshold of controllability requirement N_0;
Output: The improved adjacency matrix A';

```
while N_d > N_0 do
    Compute the eigenvalues of A and determine λ_M with maximum multiplicity;
    Implement elementary row transformations on the column vectors of λ_M I_N + A and get the row canonical form C_A, let k = rank(C_A), N_d = N − k
    Determine a set of driver nodes S_D;
    Select two different target nodes t and s from S_D ∩ {k+1, k+2, ···, N} such that A(s,t) = 0;
    if Link s ↔ i is not linkable then continue;
    end if
    for i = 1; i < N; i++ do
        if Link t ↔ i is deletable then A(t,i) = 0, A(i,t) = 0;
        end if
        if Link s ↔ i is deletable then A(s,i) = 0, A(i,s) = 0;
        end if
    end for
    Let A(s,t) = 1, A(t,s) = 1;
end while
return A' = A;
```

statement "deletable" and "linkable" in Algorithm 1. Exact controllability can never be the unique concern to a WDC architecture designer. One single link many be redundant to controllability but meanwhile crucial to some other performance indicators. Thus deleting it arbitrarily may cause intolerable performance loss. Then we consider this link as "undeletable". Otherwise, if it is not important for any concerned indicators, then it is "deletable". For the statement "linkable", adding a new link to the network is also facing the same issue. Besides, there may be some other restrictions need to be considered before linking any two nodes in the network. In a word, "joint optimization" is the most important rule for any designers who want to improve the controllability of their WDCs.

We give a brief proof on the effectiveness of Algorithm 1. It suffices to show Algorithm 1 can extend this column vector group and maintain its linear independence. We denote one of such maximal linearly independent group by $\{p_1, p_2, \cdots, p_k\}$, the rest column vectors of B by $\{p_{k+1}, p_{k+2}, \cdots, p_N\}$, which corresponds to the driver nodes. Suppose that we select p_i and p_j by implementing Algorithm 1 once, it implies that $p_{ij} = p_{ji} = 0$, and $i > k, j > k$, where p_{ij} is the j-th component of vector p_i. By a series of elementary row operations, say $R_1, R_2, \cdots, R_m$, where $R_s, 1 \leq s \leq m$ is row interchange, row scaling or row addition, B is reduced to row canonical form. This procedure can be formulated as follows.

$$R_m \cdots R_2 R_1 (p_1, \cdots, p_k, p_{k+1}, \cdots, p_{N-k}) = \begin{pmatrix} I_k & * \\ \mathbf{0} & \mathbf{0} \end{pmatrix} \tag{6}$$

We adjust the topology by letting $p_{ij} = p_{ji} = 1$, and denote the vector by p_i' derived from adjusting p_i. Then $p_i' = p_i + e_j$, where e_j is the j-th unit vector. By matrix theory, elementary row operations do not change the linear independence of column vectors, therefore, we only need to show that p_i' is linear independent to $\{p_1, p_2, \cdots, p_k\}$. Consider that p_i is linear dependent to $\{p_1, p_2, \cdots, p_k\}$ and $p_i' = p_i + e_j$, it suffices to prove that e_j is linear independent to $\{p_1, p_2, \cdots, p_k\}$. Since B is symmetric, we rewrite the result as

$$B' = \begin{pmatrix} B_0 & C \\ C' & D \end{pmatrix} \tag{7}$$

where B_0 and D are symmetric and C' is the transpose of C. Due to the column vectors of C are linear dependent to the column vectors of B_0 as hypothesis, then the row vectors of C' are linear dependent to the row vectors of B_0 because $B_0' = B_0$. Thus, B_0 is $k \times k$ a full rank matrix. Suppose that $t_1 p_1 + t_2 p_2 + \cdots + t_k p_k = e_j$, by considering the first k rows, $(t_1, \cdots, t_k)B = \mathbf{0}$, we get $t_1 = t_2 = \cdots = t_k = 0$, thus e_j is linear independent to $\{p_1, p_2, \cdots, p_k\}$. Therefore Algorithm 1 can effectively enlarge the rank of B, thus increase the controllability of A.

Now we further discuss the complexity of Algorithm 1. The space complexity involves storage cost mainly taken by two $N \times N$ matrices A and B, which is on the order of $O(N^2)$. In each iteration of Algorithm 1, computing the eigenvalues of A requires time on the order of $O(N^3)$, elementary row transformations require time on the order of $O(N^2(\log N)^2)$, other operation requires no more than $O(N^2)$, thus the time complexity of one iteration is on the order of $O(N^3)$. Suppose that we implement t iterations in our adjustment, the overall time complexity is $O(tN^3)$. This time complexity is a little high, but it is still tolerable for local topology optimization.

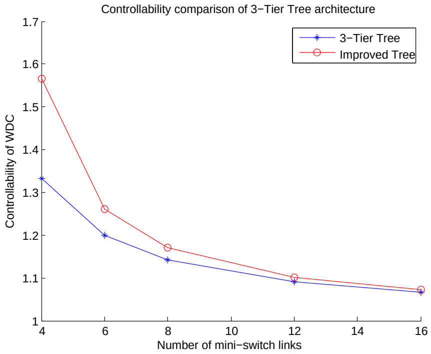
It is worth mention that even though this algorithm is applicable to improve the controllability of any network topology, not limited to WDCs, we still restrict its application in WDCs. The reasons are as follows: (1) Data center networks are moderate in scale for topology adjustment. An internet-wide network is too huge to highlight the effectiveness of local topology adjustment, while the similar operation on data center networks would be much more significant; (2) The cost of topology adjustment of WDCs is low enough in implementation, while alternations of cabling on wired network is much more expensive.

4.2 Experimental Results

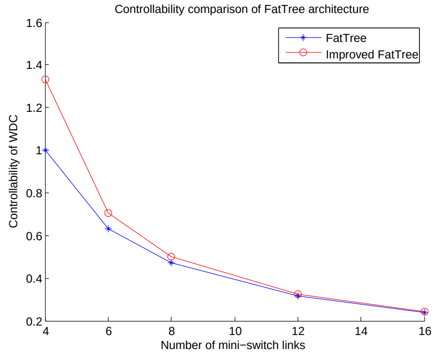
To verify the effectiveness of our method, we test Algorithm 1 on the aforementioned WDCs with d-link switches, where $d = 4, 6, 8, 12, 16$ respectively. We implement $\frac{d}{2}$ iterations of adjustment accordingly. To avoid any negative impact on other indicators of these WDCs, we adjust the topology only by adding one link in each iteration without deleting any links. The experimental results are illustrated in Fig. 8. As d increases, the scale of networks become larger quickly, the original controllability of WDCs decrease except Clos, which firstly increases then decreases. The most non-significant on controllability improvement is reflected in 3-Tier Tree (Fig. 8(a)) and FatTree (Fig. 8 (b)). The controllability n_D of 3-Tier Tree increases from 1.33 to 1.56 for 4-link switch cases, indicates a promotion of 17.3%. While the n_D of FatTree increases from 1 to 1.33, indicates a promotion of 33%. That's because the server nodes are much less than other WDCs in proportion. The marginal benefits quickly decrease along with d increases. For Clos and Bcube architectures, the performance gets even better, achieves a 100% and 66.9% promotion, respectively (Figs. 8 (c) and (d)). For ease of comparison, we set the number of adjustment iterations to be $\frac{d}{2}$. Adding only $\frac{d}{2}$ links is a very slight change on the whole WDCs, it costs almost nothing in wireless networks, however the enhancements of controllability are considerable, especially for Dcell and FiConn, which originally perform better than other WDCs. If topology designers want to get better n_D performance on 3-Tier Tree and FatTree, he needs only implement more adjustment iterations, for instance, $3d/4$, d, even $2d$, or some other specified number. Adequately control efficiency needs more adjustments on the original topology. In application, the number of iterations in implementing Algorithm 1 depends on the controllability requirement of topology designers.

5. FUTURE WORK AND CONCLUSIONS

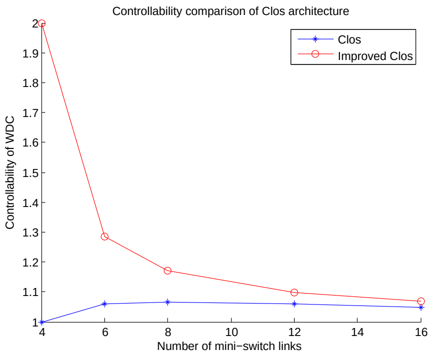
In this paper, we firstly proposed a novel concept that controllability should be considered as an important performance indicator of WDCs. Based on exact controllability theory, we compared the controllability of several mainstream WDC architectures and pointed out that Dcell_1 and FiConn_1 are of much better performance in controllability than other architectures. Then we deeply analyzed the decisive factors to controllability and proposed a feasible method to improve the controllability of WDCs. We verified our method by improving the controllability of actual WDC architectures with different number of switch links, the experimental results showed that the controllability can be significantly improved even by adding a small proportion of links to the original topology. For ease of specification, our analysis is limited on several simple instance of WDCs, some newly designed architectures for WDCs are not investigated. However, our method is universally applicable in improving the controllability of any WDC architectures with wired or wireless links in their physical topology. Besides, the quantitative study and comparison of universal cases of WDCs, such as Dcell_k may be a future work.



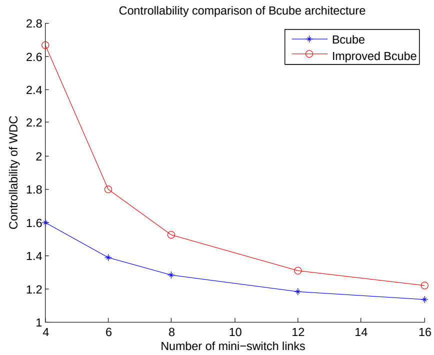
(a) 3-Tier Tree



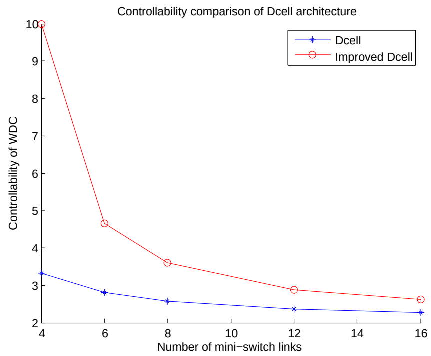
(b) FatTree



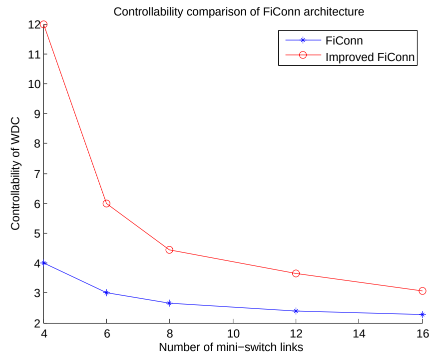
(c) Clos



(d) Bcube



(e) Dcell



(f) FiConn

Fig. 8. Experimental results of controllability adjustment on mainstream WDC architectures. The X-axis is d, the number of links of switches in the WDCs, the Y-axis is n_D, the corresponding controllability. The blue lines marked by '*' stand for the original controllability of WDCs, while the red lines marked by 'o' shows the improved results. Note that the scale ranges of Y-axis vary in different subfigures.

ACKNOWLEDGMENT

This research is supported by the National Natural Science Foundation of China (NSFC) under Grant Nos. 61871048 and 61872253.

REFERENCES

1. Z. Zhang, "HSDC: A highly scalable data center network architecture for greater incremental scalability," *IEEE Transactions on Parallel and Distributed Systems*, Vol. 30, 2019, pp. 1105-1119.
2. B. Wang, "A survey on data center networking for cloud computing," *Computer Networks*, Vol. 91, 2015, pp. 528-547.
3. M.-H. Tsai, "Enabling efficient and consistent network update in wireless data centers," *IEEE Transactions on Network and Service Management*, Vol. 16, 2019, pp. 505-520.
4. A. S. Hamza, "Wireless communication in data centers: A survey," *IEEE Communications Surveys and Tutorials*, Vol. 18, 2016, pp. 1572-1595.
5. C. M. Schneider, "Mitigation of malicious attacks on networks," *PNAS*, Vol. 108, 2011, pp. 3838-3841.
6. P. Xiao, "Detecting DDoS attacks against data center with correlation analysis," *Computer Communications*, Vol. 67, 2015, pp. 66-74.
7. A. Bhardwaj, "Three tier network architecture to mitigate DDoS attacks on hybrid cloud environments," *ACM Computing Surveys*, Vol. 67, 2015, pp. 66-74.
8. Y.-C. Wang and S.-Y. You, "An efficient route management framework for load balance and overhead reduction in SDN-based DCNs," *IEEE Transactions on Network and Service Management*, Vol. 15, 2018, pp. 1422-1434.
9. U. Zuneera, "An efficient wireless control plane for software defined networking in data center networks," *IEEE Access*, Vol. 7, 2019, pp. 58158-58167.
10. S. R. Villarreal, "Optimizing green clouds through legacy network infrastructure management," in *Proceedings of the 13th International Conference on Networks*, 2014, pp. 142-147.
11. X. Hu, "Joint workload scheduling and energy management for green data centers powered by fuel cells," *IEEE Transactions on Green Communications and Networking*, Vol. 3, 2019, pp. 397-406.
12. M. Noormohammadpour and C. S. Raghavendra, "Datacenter traffic control: Understanding techniques and tradeoffs," *IEEE Communications Surveys and Tutorials*, Vol. 20, 2018, pp. 1492-1525.
13. T. Chen, "Pache: A packet management scheme of cache in data center networks," *IEEE Transactions on Parallel and Distributed Systems*, Vol. 30, 2019, pp. 1-14.
14. Y. Y. Liu, J. J. Slotine and A. L. Barabási, "Controllability of complex networks," *Nature*, Vol. 473, 2011, pp. 167-173.
15. J. Gao, "Target control of complex networks," *Nature Communications*, Vol. 5, 2014, pp. 5415:1-8.
16. Z. Yuan, "Exact controllability of complex networks," *Nature Communications*, Vol. 4, 2013, pp. 2447:1-9.

17. C. E. Leiserson, "Fat-trees: Universal networks for hardware-efficient supercomputing," *IEEE Transactions on Computers*, Vol. C-34, 1985, pp. 892-901.
18. C. Clos, "A study of non-blocking switching networks," *Bell System Technical Journal*, Vol. 32, 1953, pp. 406-424.
19. C. Guo, "Bcube: A high performance, server-centric network architecture for modular data centers," in *Proceedings of ACM SIGCOMM Conference on Data Communication*, 2009, pp. 63-74.
20. C. Guo, "Dcell: a scalable and fault-tolerant network structure for data centers," in *Proceedings of ACM SIGCOMM Conference on Data Communication*, 2008, pp. 75-86.
21. D. Li, "Ficonn: Using backup port for server interconnection in data centers," in *Proceedings of IEEE INFOCOM*, 2009, pp. 2276-2285.

Yi Zhang received the M.S. degree in Electronics from Mid Sweden University, Sweden. She is currently pursuing her Ph.D. degree in Institute of Network Technology, Beijing University of Posts and Telecommunications. Her research interests include network security, architecture design and robustness analysis of network topology, network controllability, *etc*.

Tengfei Cao received his M.S. degree in Computer Science from Chongqing University in 2012. He is currently working towards the Ph.D. degree with the Institute of Network Technology, Beijing University of Posts and Telecommunication BUPT. His research interests include multimedia communications over 5G networks.