NSDI Paper Revision Plan

Thanks for all the valuable reviews. We summarize all the concerns below (the order of the concerns may not follow the sequence of the reviews). We plan to address all the comments one by one as follows:

Main Concerns:

1. The reviewers determined that the paper needed to have an improved cost model. In particular, the reviewers wanted the paper to consider the costs of cables between a normal fat tree and a Folded Clos. As needed, it is okay to make reasonable assumptions about rack size and placement.

Thanks for this suggestion. We will compare cable costs with the following assumptions in the revision:

1) We will assume that the price/meter is $25 for electrical cables (https://www.fs.com/c/100g-dac-aoc-2873) and the price/meter is $5 for single-mode fiber optic.
2) We will assume that intra-rack connections use short-range electrical cables, while inter-rack connections use optical cables. Note that, when optical cables are used, the cost of optical transceivers must be included and we assume that each optical transceiver costs 200$.

We expect that the cable cost of FC is higher than that of Clos built using the same number of electrical switches. Nevertheless, we believe that this part of cost only accounts for a small fraction of the overall network cost.

2. The reviewers also wanted some more issues in the paper to be addressed. For example, Figures 7, 8, and 9 are not referenced anywhere.

Admittedly, there are some typos and confusions in our paper to be addressed. We will address them one by one.

1. *Figures 7, 8, and 9 are not referenced anywhere.*

Figure.7 has not been referenced when we describe and analyze the throughput of different routing policies, but Figure.8 and 9 have been referenced when we analyze the contents in the relative figures.

For Figure.7, In section 4 “Numerical Throughput Analysis”, the detailed analysis for the results has been presented and we will add references for Figure.7 at proper locations in section 4. For references of other figures like Figure.8 and Figure.9, we will check if the references are proper or not.

2. *Some references in the text point to incorrect figures. For example, “As shown in Fig. 13(a) and 13(b), FC attains 1.1−2× the throughput of Clos networks.” – Figures 13 (a) and (b) actually show*

CDFs of FCTs.

In this part, we have referenced Figure.13 by mistake, we will change it to Figure.12. For other references, we will check them meticulously and revise them.

3. *In Figure 7a, where is the ECMP line?*

In Figure.7a, the ECMP line overlaps with the KSP-32 line, we will add a mark to distinguish them.

4. *“Note that there is a decrease in throughput when the network size changes from 700 to 1400. The reason is that constructing a two-layered Clos becomes infeasible as the number of servers increases to 1400.” – There are multi-tier Clos networks that scale to much larger sizes than 1400 servers. See, for example, “Jupiter Rising” from Google for some of the deployed ones. It would have been great to include some of those topologies.*

Sorry for the confusion. In this part, we want to explain why the throughput performance decreases. Because two-layered Clos cannot scale beyond 1400 servers in this case, we have to build a three-layered Clos to support 1400 servers. The statement may let others misunderstand that our Clos network in the experiment can’t scale network to large sizes.

We will clarify that we build three-layered Clos for larger network in this part to avoid misunderstanding.

5. *There are a bunch of grammar errors, e.g., "the PFC-induced deadlock cannot go away once occurs even if we restart all the servers." --> "the PFC-induced deadlock cannot go away once it occurs even if we restart all the servers."*

For the grammar errors, we will check the whole paper and correct all of them.

6. *There is a typo in the definitions in Section 3.1: "Each switch has p = s + h ports, with p of which connected to the hosts and s of which con- nected to other ToR switches." should be "Each switch has p = s + h ports, h of which connect to the hosts and s of which connect to other ToR switches."*

We will check and revise the definition errors in the paper.

7. *It was surprising to see that this paper does not cite "Flattened butterfly: a cost-efficient topology for high-radix networks" from ISCA '07 given that it seemed like the title of this paper was inspired by that paper.*

We did borrow the idea from “flattened butterfly” while we name our design. We will add the reference.

8. *It is not correct to call citation [39] TCP-Bolt.*

We will change “Tcp-Bolt” to DF-EDST, which is used in [39].

[39] Stephens B, Cox A L. Deadlock-free local fast failover for arbitrary data center networks[C]//IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications. IEEE, 2016: 1-9.

3. The reviewers also wanted the paper to include a discuss significant prior work: "Detecting

and Resolving PFC Deadlocks with ITSY Entirely in the Data Plane", INFOCOM '22. In particular, this paper needs to justify why a Flattened Clos network is needed if there exist some switches where deadlocks can be broken at the switch level.

This paper (ITSY) designed three data plane mechanisms and realized them on P4 switches to prevent deadlocks. However, when concurrent deadlocks happen, two of the three mechanisms including Port-level resolution and Flow-level resolution may experience unexpected interference without an extra priority operation. Only the count-min based drop mechanism works, but this mechanism may cause packet drop, which may significant hurt RDMA performance. In summary, the data-plane mechanisms in ITSY are not ideal yet.

Even if we have an ideal data-plane mechanism to resolve deadlocks, the P4 switches may not work as expected due to unexpected hardware or software bugs. Although control-plane mechanisms can be used to break deadlocks, such mechanisms are much slower than data-plane mechanisms. Therefore, we still need FC as a backup plan. Although FC's virtual up-down routing offers lower throughput than the KSP routing, FC's virtual up-down routing guarantees no deadlock.

We will add the above discussion in the paper.

Response to other concerns:

We summarize other concerns as follows and will address them as well.

1. Even if the routing policy like KSP may lead to CBD, the possibility that the CBD will turn into the Deadlock may be small under the circumstances that the network can't be under the extreme cases like the experiments in the paper. What's more, choosing the port to the nodes that are not on the loop can also decrease the possibility of Deadlock. So is it meaningful to design such policy without the best throughput performance to solve the deadlock?

Although the deadlock possibility is small, when a data center runs for a long time, deadlock can still be triggered and cause severe consequences. Hence, it is crucial to design a deadlock-free policy.

We will clarify this point in the paper.

2. Is it practical to use OCS to reconfigurations to address the worst cases? (1) Traffic can change very unpredictably and this may lead to too many reconfigurations? So maybe it is unrealistic to configure the OCS too frequently. (2) Which kind of topology should be transferred? Maybe some guidelines are needed. (3) The consideration of using optical switches to reducing wiring seems to be missing a larger optimization problem in designing a hybrid optical/electrical data center network topology.

The reviewer actually raises a promising future direction: optimize topology based on traffic patterns to attain better network performance for RDMA traffic. This is indeed one of our next steps. Our previous work [1,2] and Google's Jupiter data center [3] have shown that low-frequency topology engineering is feasible, and reconfiguring OCSs for every traffic change may not be necessary. We suspect that this conclusion still holds for lossless RDMA networks.

As for how to compute traffic-aware topologies**,** we can generate the topology based on the history traffic patterns. One possible solution could be a) generating a number of candidate FC topologies; b) and selecting one with the lowest average hop count for different historical traffic patterns. There are still more guidelines that can be explored in the future work. We leave those investigations to the future work.

We will add the above discussion in the paper.

[1] Peirui Cao, Shizhen Zhao S, Min Yee Teh, et al. TROD: Evolving from Electrical Data Center to Optical Data Center[C]//2021 IEEE 29th International Conference on Network Protocols (ICNP). IEEE, 2021: 1-11.

[2] Min Yee Teh, Shizhen Zhao, Peirui Cao, Keren Bergman, "Enabling Quasi-static Reconfigurable Networks with Robust Topology Engineering," to appear in IEEE/ACM Transactions on Networking 2022.

[3] Poutievski L, Mashayekhi O, Ong J, et al. Jupiter evolving: transforming google's datacenter network via optical circuit switches and software-defined networking[C]//Proceedings of the ACM SIGCOMM 2022 Conference. 2022: 66-85.

3. The paper does not consider the failure handling in detail. It will be stronger if it puts forward a proper way to handle that considering failures usually happen in the Datacenter.

Failure handling is a big topic for data centers. We have a discussion on failure handling in Section 6. We will explore it more in our future work.

4. The buffer of the traffic has increased a lot, from 10MB to over 100MB now. So it may be possible to assign different lossless priorities for packets at different hops? The lossy RDMA is supported by Mellanox ConnectX-4 onwards NICs, and Mellanox ConnectX-4 onwards NICs have been widely used. Maybe lossy RDMA may provide us with simpler ways? Besides, what about taggers to prevent Deadlock?

(1) Though the switch buffers have increased a lot, the data center link bandwidth has been increasing much faster and the buffer/bandwidth ratio is actually decreasing over time. Hence, supporting more lossless priorities will become more difficult. As a result, the idea to assign different lossless priorities for packets at different hops may be not feasible.

We will reference the paper below to explain the trend in our paper.

[4] Goyal P, Shah P, Zhao K, et al. Backpressure Flow Control, NSDI 2022.

(2) Lossy RDMA completely eliminates the possibility of deadlocks. But the consequence is that those packets lost will experience longer delay. As a result, lossy RDMA may not be able to substitute lossless RDMA in all cases. Therefore, having the deadlock-free solution such as FC is still important.

(3) Tagger may require too many lossless queues to prevent deadlocks. Note that commercial data center switches usually can only support 2 or 3 lossless queues. As shown in the Tagger paper,

to eliminate deadlocks for all the shortest paths in a 2000-node jellyfish topology, 3 lossless queues are required. If we use 32-way KSP routing, which has about O(10)xO(1000)xO(1000)=O(10^7-10^8) more paths than the shortest paths in the same 2000-node jellyfish topology, many more lossless queues would be required.

We will add the above discussion in the paper.

5. the probability of CBD is based on theoretical analysis. can we show some evidence with emulation or real-world data? how likely would it happen in most cases?

We believe that it is sufficient to analyze the probability of CBD based on theoretical analysis. In practice, even though the probability of triggering deadlocks may not be high in a network with CBD, the impact of a deadlock can be catastrophic. We will emphasize this point in the paper.

6. A weakness of this paper is that the scope of the problem being solved is somewhat narrow. Similarly, deploying a new topology is difficult, and it is not clear that the Flattened Clos will see adoption in practice.

Google has deployed a layer of optical circuit switches in their data centers [3], which makes it possible to reconfigure topologies on demand. We believe that it is easy to apply FC in a data center with topology reconfiguration capability.

7. The paper does not precisely enough compare FC and Clos topologies even by the number of switches. The paper says that "we choose a Clos net-work that offers the maximum throughput to the H servers using roughly the same number of switches.", and this is not precise enough.

We have described how to generate a Clos network in Appendix A.3 in details. We will add some additional numbers in the revision.

8. This paper would have been improved if it had considered using virtual channels to increase path diversity. How much could this potentially increase throughput by?

We believe that the reviewer pointed out a promising future direction. FC uses only one lossless queue, and its throughput performance is slightly lower than that of the KSP routing. A RoCEv2 data center can support two or three lossless queues. If we let FC use more than one lossless queues, more paths can be utilized and the throughput could see improvement. We will explore it further in the future.

9. The load and location for Figure.13 may be improper. First, for expander graph, is the load used in the Figure.13 relatively low? Maybe it's better to use higher load and some of the real traces. Besides, the performance comparisons among Disjoint up-down, Clos and Edst may be important part in the evaluation. Maybe it is improper to put them in appendix.

As described in the paper below, the median link utilization varies between 10% to 20% and the busiest 5% utilization of links is between 23% to 46%. So, a 30% network load is realistic in real-world Datacenters. We will add an explanation in the paper.

As for the locations, we plan to exchange the locations between Table.2 and some results of Figure.13.

[5] Roy A, Zeng H, Bagga J, et al. Inside the social network's (datacenter) network[C]//Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication. 2015: 123-137.

10. Finally, the scalability of the routing tables can be worried about. The complexity of the algorithm would prevent it from running on the fly so that entries need to be stored on each ToR as, unlike Clos, there does not seem any easy way to aggregate IP address given the way the topology is wired?

The reviewers actually raise a great question for all expander graphs, including Jellyfish, xpander, FC, etc. One potential idea is to group ToRs, and perform hierarchical routing. This could introduce some opportunity to aggregate IP addresses. We will study this problem in greater detail in our future work.

11. The throughput results in Figure 12 are somewhat inconclusive as it is not clear how representative are the all-to-all and uniform random traffic patterns vs. the near-worst pattern for today's workloads. It would have been helpful if the authors would have complemented these more abstract scenarios with more realistic published traces.

We think the three traffic patterns are representative when evaluating the throughput performance. The worst pattern can tell us the minimum throughput of a network topology and the network topology with better performance can be more robust to traffic uncertainty. Paper [6] has used the near-worst pattern to evaluate the performance of Datacenter. Besides, based on the results of [7], if the throughput performance of all-to-all traffic pattern is x, the worst performance for any traffic patterns is at least x/2. Therefore, understanding the performance of the all-to-all traffic pattern is also important. The uniform-random pattern is chosen to approximate the traffic pattern in real world. In practice, the traffic patterns of Google's Jupiter data center are approximately uniform random (see Appendix C in [3]).

Based on the above discussion, we believe the evaluations of the three patterns can be convincing. We will add the above discussion in the paper.

[6] Namyar P, Supittayapornpong S, Zhang M, et al. A throughput-centric view of the performance of datacenter topologies[C]//Proceedings of the 2021 ACM SIGCOMM 2021 Conference. 2021: 349-369.
[7] Jyothi S A, Singla A, Godfrey P B, et al. Measuring and understanding throughput of network topologies[C]//SC'16: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis. IEEE, 2016: 761-772.

12. To be clear, you can still have a Cyclic Buffer Dependency (CBD) loop but in some of those

nodes in the loop, you also have other options for sending out the traffic to a port other than the nodes on the loop. This should significantly reduce the probability of a theoretical CBD in the graph turning into an actual deadlock. The key requirement for forming a deadlock – the one-to-one links – is no longer there. The paper does not at all discuss this. It simply states that "Under ECMP or KSP routing, it is still possible to have four flows taking the paths." Sure, but those flows will also have other options and when a port is locked the switch should be able to forward the flow through an unlocked one, breaking the cycle and preventing the formation of deadlock.

The reviewer actually offers a high-level idea to avoid deadlocks. If we implement this idea in the control plane, the response time can be too slow. If we implement this idea in the data plane, we may need hardware support like P4 and design some deadlock-prevention mechanisms like ITSY. As discussed in "Main concerns.3", the data-plane deadlock-prevention mechanisms are not mature yet, and FC can be still useful.

We will add some discussion in the paper.