Applying Traffic Merging to Datacenter Networks

Prof. Haresh R. Parmar

Assistant Professor in Computer Engineering Department @ Silver Oak College of Engineering & Technology

Prof. Nilesh Solanki

Assistant Professor in Computer Engineering Department @ Silver Oak College of Engineering & Technology

Abstract

The problem of reducing energy usage in datacenter net-works is an important one. However, we would like to achieve this goal without compromising throughput and loss char-acteristics of these networks. Studies have shown that data- center networks typically see loads of between 5% {25% but the energy draw of these networks is equal to operating them at maximum load. To this end we examine the problem of reducing the energy consumption of datacenter networks by merging trance. The key idea is that low trance from N links is merged together to create K _ N streams of high trance. These streams are fed to K switch interfaces which run at maximum rate while the remaining interfaces are switched to the lowest possible rate. We show that this merging can be accomplished with minimal latency and energy costs (less than 0.1W total) while simultaneously allowing us a deterministic way of switching link rates between maximum and minimum. We examine the idea of trance merging using three deferent datacenter networks {attended buttery, mesh and hypercube networks. In addition to analysis, we simulate these networks and utilizing previously developed trance models we show that 49% energy savings are obtained for 5% per-link load while we get 20% savings for a 50% load for the attended buttery and somewhat lower savings are obtained for the other two networks. The packet losses are statistically insignificant and the maximum latency increase is less than 3_s. The results show that energy-proportional datacenter networks are indeed possible. Data network is a telecommunications network that allows computers to exchange data. In computer networks, networked computing devices pass data to each other along data connections. The connections (network links) between nodes are established using either cable media or wireless media. The best-known computer network is the Internet. Network computer devices that originate, route and terminate the data are called network nodes.

Nodes can include hosts such as personal computers, phones, servers as well as networking hardware. Two such devices are said to be networked together when one device is able to Exchange information with the other device. Whether or not they have a direct connection to Each other. Computer networks support applications such as access to the World Wide Web, shared use of application and storage servers, printers, and fax machines, and use of email and instant messaging applications. Computer networks differ in the physical media used to transmit their signals, the communications protocols to organize network traffic, the network's size, topology and organizational intent.

Introduction

The electricity consumption of datacenters is a significant contributor to the total cost of operation over the lifetime of these centers and as a result, there have been several studies that aim to reduce this cost. Since the cooling costs scale as 1.3x the total energy consumption of the datacenter hardware, reducing the energy consumption of the hardware will simultaneously lead to a linear reduction in cooling costs as well. Today the servers account for around 90% of the total energy costs, regardless of loading. However, since typical CPU utilization of server clusters is around 10□50% [1], there are several efforts underway to scale the energy consumption of the servers with load. It is expected that in the near future, sophisticated algorithms will enable us to scale the energy consumption of the servers linearly with load. When this happens, as noted in [1], the energy cost of the network will become a dominant factor. Hence, there is significant interest in reducing the energy consumption of the datacenter networks as well. Various authors [1, 2, 3] note that the average trance per link in deferent datacenter networks tends to range between 5% and 25%. To save energy, the authors in [1] implement a link rate adaptation scheme, whereby each link sets its rate

Every 10 □ 100 _s based on trance prediction. The energy savings are shown to be 30□45% for deferent workloads for loads less than 25%. However, the scheme supers from the problem of packet losses due to inaccurate trance prediction as well as significantly increased latency. Indeed, the mean increase in latency is between 30 □ 70 _s for deferent loading scenarios. Other general approaches attempt to reduce network-wide energy consumption by dynamically adapting the rate and speed of links, routers and switches as well as by selecting routes in a way that reduces total cost [4, 5, 6]. In this respect, these green networking approaches have been based on numerous energy-related criteria, applied to network equipment and component interfaces [5, 6]. These approaches tackle the minimization of the network power consumption by setting the link capacity to the actual traffic load. we present an innovative approach to adapt energy consumption to load for datacenter networks. The key idea is to merge trance from multiple links prior feeding it to the switch. This simple strategy allows more switch interfaces to remain in a low power mode1 while having a minimal impact on latency. We have explored the idea of trance merging in depth in the context of enterprise networks in [7, 8, 9], where we show that savings in excess of 60□70% are obtained with no aspect on trance. Indeed, the big advantage of the merge network is that, unlike the most other approaches, it works in the analog domain, so it does not introduce delays for store-and-forward Layer 2 (L2) frames, rather it redirects such frames on-the-y at Layer 1 (L1) between external and internal links of the merge network itself. In addition, the merge network allows reducing frequent link speed transitions due to the use of the low power mode. In our approach, such transitions happen only infer quaintly thus allowing us to minimize the delay due to the negotiation of the new link rate and the additional energy required for the rate transition. In this paper, we apply the merge network concept to three deferent datacenter network topologies { Flattened Buttery [1, 10], Mesh and Hypercube [1, 11]. Using extensive simulations we then show that up to 20% □ 49% energy savings are possible for loads between 50% and 5% respectively for the attended buttery and somewhat lower savings for the mesh and hypercube. The rest of the paper is organized as follows. The next section discusses the concept of trance merging. The subsequent section describes the deferent datacenter network topologies we study and in Section 4 we present a theoretical model of energy savings when the merge network is applied to these topologies. Section 5 then presents our simulation methodology and results. Finally, our conclusions are presented in Section 6.

SYSTEM MODEL

Figure 1 illustrates the usage model we consider. In general, we assume there are *n* pairs of communicating nodes and some number *k* of repeaters deployed about the room (the repeaters may well be other idle nodes that are tasked to aid active connections). All the nodes and repeaters are assumed to be equipped with smart antennas, each with *M* antenna elements. The nodes and repeaters can beam form in any direction. Further, since a repeater serves to connect a communicating pair of nodes, we assume that it can simultaneously communicate with both the nodes that form the end-points of the link. Thus the repeaters may be implemented either as store and forward nodes that receive packets on one link and then forward them on the other or as cut-through devices where the incoming signal is not decoded but simply forwarded on the outgoing link. We note that the analysis in this paper is valid for either model. The problem we consider can be summarized as follows: given *n* communicating node pairs and *k* repeaters, how can we establish *n* connections such that data rates are maximized for each pair? The problem is non-trivial because of interference and the existence of obstructions in the LoS path between pairs of communicating nodes.

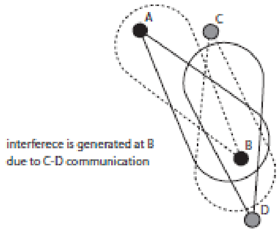
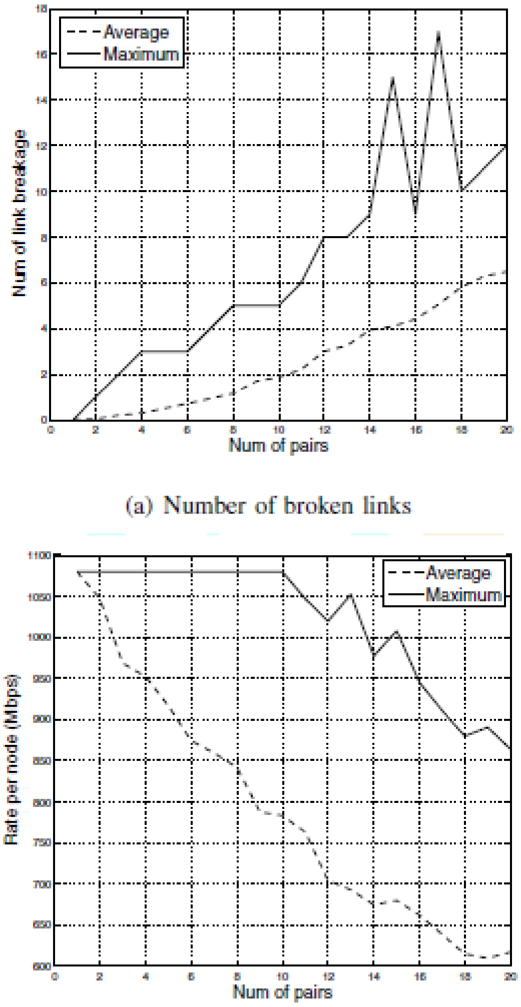


Fig. 3. Degradation of a link due to interference from another.

Figure 3 illustrates a simple case where one link interferes with another, thus reducing the data rate for that link1. As we can see, the transmissions from node A to B will generate an interfering signal at node D thus degrading the SINR (Signal to Interference and Noise Ratio) and the data rate at D. One can argue that with narrow enough beams, the amount of such interference can be eliminated or made negligible. In order to study this assertion further, we ran Mat lab simulations for random node placements and measured the interference. For a given *n*, we randomly uniformly place each node of that link somewhere within a room of size 10mx10m. Each node is assumed to have a *linear array* with *M* = 20 antenna elements. We use standard expressions for computing the array factor (AF) [7], The more realistic version of the problem is one where *each pair* of communicating nodes tries to optimize its performance independently of the other pairs. In this *distributed* version of the problem, the definition of optimum remains unchanged but the problem of finding the optimal solution is harder. In the next section we focus on this distributed problem and develop a simple solution to it.



(a) Number of broken links

(b) Throughput

Fig. 4. Impact of interference.

GREEDY ALGORITHM

We develop a distributed greedy algorithm for finding an allocation that, in most cases, achieves the optimal link allocation. The algorithm is iterative and works as follows: 1) Initially each link is set up directly between the two end-points. 2) Each pair computes the best achievable rate for each direction of communication. 3) If a link does not achieve 800Mbps rate (in either direction), it will randomly uniformly choose a free repeater. 4) The link is now set up via this repeater, and this is done by all the links that fall below 800 Mbps. 5) After this step, each link recomputed the achievable data rate. It is possible that a previously good link now shows degraded performance due to interference from a newly rerouted link. As previously, every link that falls below the 800 Mbps threshold selects a new free repeater. 6) The algorithm iterates until no further improvement is seen in two consecutive steps. It is possible that the algorithm terminates with some pairs seeing data rates that are below the 800Mbps threshold. Figure 5 illustrates the workings of this algorithm for a case when we have n = 6 and k is unrestricted (this is a screenshot of a visualization tool built on top of our Mat lab simulator). The room is 10mx10m and all nodes as well as repeaters are at a height of 1m. In the figure, each of the six pairs is labeled 1– 6 and the repeaters that get used are numbered R1, R2, etc. Initially, each of the pairs sets up a direct connection between the two end-points using their smart antennas. The bottom two bar charts in the figure correspond to the *four* iterations of the algorithm where the SINR and Rate is shown at the end of each iteration for each of the six pairs of nodes. Each bar (in a set of four bars) is one iteration of the algorithm for a given node. In the figure, we plot the minimum observed SINR for each pair as the first bar (of the four bars) and is labeled by ‘D’ (this is the direct path). The achieved data rate for each of the pairs is shown in the bottom most plot. Pairs 3 and 4 have a low data rate of 600Mbps and they each re-route the connection via repeaters in the next step – pair 3 goes via R8 and pair 4 goes via R1. The new SINRs and data rates for the five pairs are shown as the second bar in each group of bars in the bottom two plots. As a result of this re-routing (pair 3 via R8 and pair 4 via R1), the SINR for pair 5 drops, as does its data rate. Pairs 1 and 2 also see a small degradation in SINR but the data rate remains high. In the next iteration, pair 4 switches from R1 to R4 and pair 5 now chooses to go via a repeater R7. This improves pair 5’s data rate but pair 4 is still below threshold. Finally, pair 4 changes the repeater yet again and selects R3. At this point, all the pairs have a data rate greater than the threshold of 800 Mbps and the algorithm terminates.

Experimental Evaluation

The goal of the simulations is to understand the effectiveness of repeaters in mitigating link failure. The *metrics* we used to study this question are: • Data rate achieved per user,

• Number of repeaters used to fix *all* link breakages,

• Percentage improvement in throughput when using repeaters.

In order to get a comprehensive understanding of how repeaters may help, we used a large number of node placements in our study. Specifically, we use a room of size 10mx10m within which we placed 2*n* nodes randomly uniformly giving us *n* links. We considered *n* = 4, 5, 6, 7, 8. For each value of *n* we randomly generated 1000 different configurations and studied the performance of our algorithm in each case. Repeaters are placed at grid locations within the room and we use 16 repeaters in all. Note that no more than *n* repeaters will be used for a given *n* since we only consider cases when a link is routed through at most one repeater. The case when the number of repeaters *k* < *n* is a subset of the case when *k* is unrestricted. For instance, if the number of repeaters used for a *n* is *l* then we know that using *k* < *l* will result in (*k*−*l*) broken links.

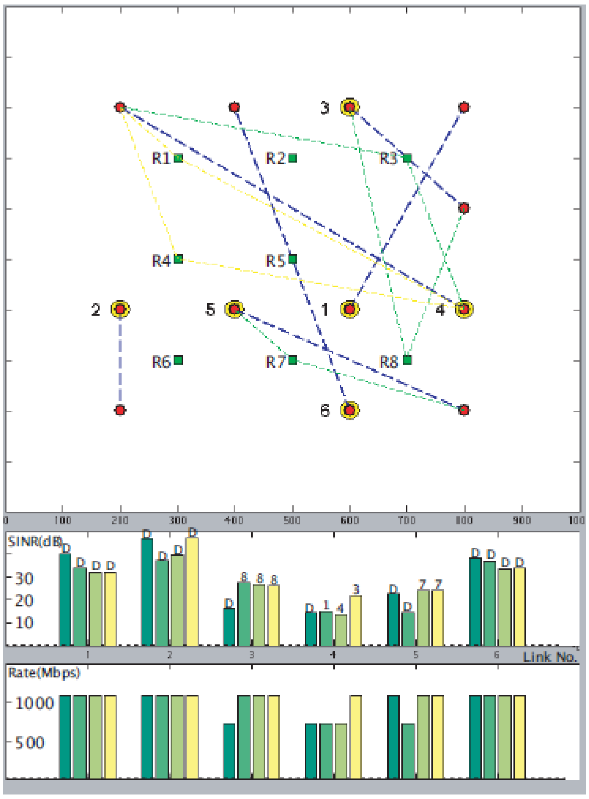


Fig. 5. Illustration of the greedy algorithm.

Finally, we place the nodes and repeaters all at a height of 1m above the floor. There are two reasons for this choice. First, in real deployments, the repeaters may actually be other idle nodes rather than special purpose devices. And second, as compared to the case when repeaters are deployed on the ceiling, the interference from repeaters towards the receivers will be significant in this case. This gives us a good lower bound on the benefits of using repeaters. In Figure 6 we plot the average per link data rate achieved as a function of the number of links with and without repeaters. We see that when using repeaters, the average per link data rate continues to be above 1Gbps whereas the data rate is much lower when we do not allow repeaters. Also, the average data rate per link falls with increasing number of links because there is greater interference, even when using repeaters. Figure7 plots the average number of repeaters used as a function of *n* (averaged over 1,000 runs). It is interesting to see that even with *n* = 8 pairs, we use an average of only 2 repeaters. But the benefits of adding these two (on average) repeaters is enormous - the average data rate jumps from less than 900Mbps to over 1Gbps/user. In order to study the application of repeaters in more detail, let us consider the case when there are *n* = 6 pairs. The plot for the data rate in Figure 8 shows the expected improvement in data rate per link when using repeaters. The x-axis reports on the number of degraded links (when all pairs use the direct LoS link). When no link is degraded the data rate seen by each pair is over 1Gbps. When one pair’s link is subject to interference, the average data rate without repeaters falls to 930Mbps. But when repaired using a repeater, the data rate climbs to over 1Gbps. When 4 or 5 of the six pairs see link degradation due to interference, the average data rate is at about 500Mbps only but jumps up to 1Gbps with repeaters. In order to understand how often links degrade, Figure 9 plots the pdf (probability density function) of the number of links that fall below threshold when repeaters are not used. 30% of the time we see that repeaters are not required since no pair sees degraded link quality. However, about 35% of the time one pair does see poor quality of its direct link. Interestingly, there are cases when 5 out of 6 links fall below threshold. This clearly underscores the impact of interference and the need for repeaters. Figure 10 plots the number of repeaters used as a function of the number of links that degrade. The interesting observation is that the number of repeaters used scales linearly with the number of degraded links – this means that in most cases, repairing one link does little to improve another link’s performance and thus each degraded link needs its own repeater. In some cases, for instance when 4 links are broken, the average number of repeaters used is 4.5. The reason for this is that re-routing a broken link via a repeater tends to break a previously good link (as we see in Figure 5 where link 5 was originally in good condition but then gets degraded due to link 4 being rerouted). Therefore, the total number of repeaters we may use could exceed the number of broken links without repeaters. In all cases, the improvement is over 50% thus, again, showing the benefits and need to use repeaters.

Switch without and with merge network.

At a switch from multiple links and feeding that to few interfaces. The motivation for doing so is the observation made by various authors that per-link

loading in datacenter networks tends to be well below 25% all the time and is frequently below 10% as well. Thus, by merging trance we are allowing several of the switch interfaces to operate in low power modes. Indeed, as we discuss in [9] it is also possible to replace high port density switches with lower port density switches without electing network performance in any way. Figure 1 illustrates the trance to/from N links are merged and fed to K interfaces. Setting the parameter K according to the incoming trance load allows us to reduce the number of active interfaces to K and enables N □K interfaces to be in low power modes. As an example, if the average trance load on 8 links coming in to a switch is 10%, we could merge all
the trance onto one link and feed it to one switch port running at maximum rate, thus allowing the remaining ports to enter low power mode. This approach divers from the more traditional approaches as in IEEE 802.3az where each link makes decisions independently about when to enter low power states. Indeed, as we will show, our approach results in almost optimal energy savings with minimal increase in latency. In order to understand how trance merging can help in datacenter networks, we need to examine the details of the merge network itself. A generic N _K merge (with K _ N) is denned with the property that if at most K packets arrive on the N uplinks (i.e. from N links into the switch) then the K packets are sent on to K sequential ports (using some arbitrary numbering system). For example, consider a 4_4 merge network as in Figure 2. a □ d denote the incoming links (from hosts2) and 1 { 4 denote the switch ports. The trance coming in from these links is merged such that trance is rest sent to interface 1 but, if that is busy, it is sent to interface 2, and so on. In other words, we load interfaces sequentially. This packing of packets ensures that many of the higher numbered interfaces will see no trance at all, thus allowing them to go to the lowest rate all the time. The key hardware component needed to implement this type of network is called selector, whose logical operation is described in Figure 3. There are 2 incoming links and 2 outgoing links. If a packet arrives only at one of the two incoming links, then it is always forwarded to the top out- going link. However, if packets arrive along both incoming links, then the earlier arriving packet is sent out along the top outgoing link and the latter

As has been stressed throughout the report, the objective of this phenomenographic research project as a whole is to gain insights in the students' learning of computer communication when taught in an internationally distributed project-oriented course. This report focuses on variations in the students' experience of network protocols, while my future work will study variations in learning in the context of the course and the interplay between their experience of learning and the context they experience. Different ways of experiencing the concept of network protocols in general as well as the three specific network protocols TCP, UDP and RMI have been identified and presented. A network protocol is, of course, understood in a context by an individual. This means that an individual experiences the protocol against the background of and interacting with a specific environment. In the analysis (see section 3.2.2) this background is stripped away; in other words, the statements made by individuals are decontextualized. The decontextualisation is an analytical tool for the researcher to draw conclusions about the distinctly different ways a phenomenon, as for example RMI, is experienced within the group. The individual statement is then, as has been described earlier, decontextualized through a dynamic process into a context at a collective level that is created by the researcher: the outcome space of the categories of description. The coming sections will explore and develop the results presented in earlier sections and related them to learning and teaching. Categories of description can only be created by the researcher for a group, at a collective level. Individuals experience particular phenomenon differently at different moments, which is to say that shifts can occur spontaneously and rapidly. With a distinction that was articulated by Pong (1999), shifts in focus can occur as inter-contextual shifts, when the context shifts, that is when a new subject is discussed, but also as intra-contextual shifts within the same context, either spontaneously by the student or as a part of a conversation. Many intra-contextual shifts have been identified in the data that forms the basis for this paper. The students in this study are advanced students in computer science in their third or fourth year, and as such they have had the opportunity to meet different views from their teachers, books etc on computer science. This might be a reason why they take different stands on various computer science issues throughout their studies. Packet along the other one. The hardware implementation, described in [7], is done entirely in the analog domain. Thus, a packet is not received and transmitted in the digital sense, rather it is switched along deferent selectors in the network much as a train is switched on the railroad. This ensures that the latency seen by a packet through the merge is minimal and the energy consumption is very small as well 3. We have also shown previously [9] that the minimum depth of an N_K merge network is log2 N + K 1 with the number of selectors needed equal to On the downlink (i.e. from the switch to the N links) the merge network has to be able to forward packets from any of the switch ports (connected to the K outputs of an N _ K merge network) to any of the N downlinks and be able to Forward up to N packets simultaneously.

Using Merge Networks In Datacenters

We propose adding the merge network to the datacenter networks in order to obtain energy savings. The manner in which we introduce the merge network into these networks is illustrated in Figure 5. We introduce a c_K1 merge between the c end-hosts connected to a switch and a separate m_K2 merge in the links connected to other switches. The reason for this separation is that the inter-switch links see more trance and may well be higher bandwidth. Thus from a hardware standpoint we do need separate merge networks. The figure shows that switch ports from K1 + 1 to c and from K2 + 1 to m are in low power mode.

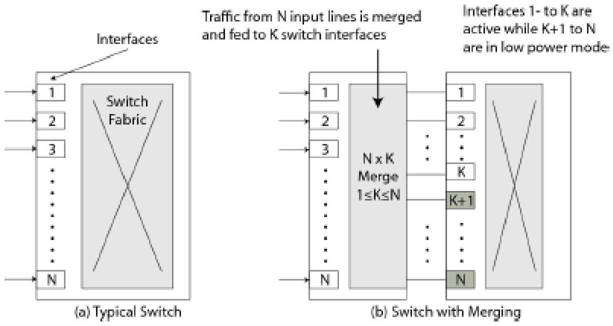

Interfaces
Traffic from N input lines is merged and fed to K switch interfaces
Interfaces 1- to K are active while K+1 to N are in low power mode
1
2
3
N
Switch Fabric
N x K Merge 1≤K≤N
K
K+1
(a) Typical Switch
(b) Switch with Merging

Figure 1: Switch without and with merge network.

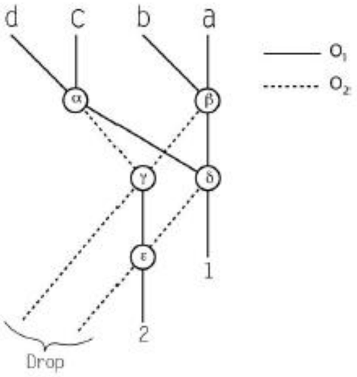


Figure 2: A 4 × 4 uplink merge network.

In order to save energy using the merge network, we need to run some number of switch interfaces at full rate while dropping the rate of the rest to the lowest possible. In other words, we need to dynamically determine the values of K1 and K2. The merge network has the unique property that links are loaded sequentially. Thus, if link i is the highest numbered active link, then in the event of an increase in load (from any or all of the hosts) the next link that will need to run at full rate will be link i + 1. This determinism in link loading gives us the key to maximizing energy savings. Specifically, the algorithm we use for changing link rates at switches is as follows:

1. if interfaces 1 to i are active (at full rate), where i is K1 or K2, then we increase the rate of the i+1th one to the full rate as well. This is done to o_set packet loss in the event of a burst of packets;

2. if at most i□2 interfaces of the i ones operating at the full rate are active, then we reduce the rate of the it interface to the lowest rate (after it goes idle).

Estimate of Energy Savings

Let us assume that the combined average load per link to and from an end-host is _. Then, the number of interfaces of the switch connected to the end-hosts that will be in active when an interface is put into low power mode, it does continue to consume energy. For example, as noted in [1], a 40 gbps interface can operate at 16 deferent rates with the lowest rate being 1.25 Gbps. The lowest rate consumes 40% of the energy of the highest rate. Thus, in computing the energy savings, we need to consider this factor as well. Let us assume that a low power interface consumes a fraction _ of a fully active interface. Then, we can write the energy savings in the interfaces when using the merge network as, what are the relevant aspects of context in data collection, if one is to maximize the variation in the pool of meaning? Starting with the researcher, the researcher acts in his or her experienced context where a particular interview is seen against the background of earlier interviews and the anticipation of the interviews to be done. That context can be seen as if the interviewee of a particular interview, through the mediation of the researcher, participates in an on-going discussion around a certain phenomenon both with the researcher and all the other interviewees, the latter being intellectually present while physically absent. The researcher has a certain aim: he wants a particular phenomenon to become the focus of mutual attention in such a way that the participant can reveal the ways in which he or she experiences it, seen from varying angles, against different backgrounds. To achieve this, he *prepares* contexts for the participants to engage with, to experience, and to speak in. In the case of eliciting written texts, this might involve devising a scenario for the participant to relate to. In holding interviews, it includes, in addition, choosing the environment where the interviews are held, choosing the theme of the interview, working out what questions to ask, Planning specific follow-up questions that might be needed, and remaining open and flexible, patient and persistent throughout.

Selecting Topology Parameters

Based on Eq. 4 and Table 1 we have a simple algorithm for selecting the value of m and hence the dimension of the network topology. For a given concentration c and maximum anticipated switch throughput we select the smallest switch and utilize its configuration that maximizes the number of interfaces. This is done because typically the marginal cost of interfaces is much smaller than the cost of the switch itself and thus it makes sense to maximize the interfaces supported. Having done this, we assign c interfaces to the end hosts and the remaining interfaces are connected to other switches giving us m. Of course, the potential drawback of being too aggressive in switch selection is that m may be too small resulting in greater network diameter and hence latency. Therefore, the particular switches selected need to

be determined also based on other network design considerations. In this paper we only concern ourselves with interface energy savings given a already denned datacenter network topology. Thus, for the remainder of the paper we only concern ourselves with Eq. 3.Figures 6 and 7 show the trend of the average interface energy savings (Eq. 3) as a function of load for varying the values of m and _. It is interesting to note that with a load less than 10% the energy saving is never less than 40% for

every possible configuration. Even with a load of 30% we are able to achieve an energy savings of more than 15%. switch and utilize its configuration that maximizes the number of interfaces. This is done because typically the marginal cost of interfaces is much smaller than the cost of the switch itself and thus it makes sense to maximize the interfaces supported. Having done this, we assign c interfaces to the end hosts and the remaining interfaces are connected to other switches giving us m. Of course, the potential drawback of being too aggressive in switch selection is that m may be too small resulting in greater network diameter and hence latency. Therefore, the particular switches selected need to

be determined also based on other network design considerations. In this paper we only concern ourselves with interface energy savings given a already denned datacenter network topology. Thus, for the remainder of the paper we only concern ourselves with Eq. 3. Figures 6 and 7 show the trend of the average interface energy savings (Eq. 3) as a function of load for varying the values of m and _. It is interesting to note that with a load less than 10% the energy saving is never less than 40% for every possible configuration. Even with a load of 30% we are able to achieve an energy savings of more than 15%.

Evaluation

In order to demonstrate the usefulness and the effective-ness of trace aggregation inside a high-performance datacenter, we evaluate the merge network using the discrete-event-driven network simulator. open-source (and free for research and educational purposes) sophisticated system used for modeling communication net- works, queueing networks, hardware architectures, and manufacturing and business processes .For our simulation, we model deferent datacenter topologies: attended buttery, mesh and hypercube. For the attended buttery, we consider an 8-ary 2-at one. For the mesh and hypercube topologies, we examine the (4,2)-ray 2 dimensional (2-D) and 8-ary 1 dimensional (1-D) cases. Hence, in each considered topology the concentration c is equal to 8 and the number of nodes is 64. We don't use over-subscription, so that every host can inject and receive at full line rate. Links have a maximum bandwidth of 40 Gbps. Switches are both input and output

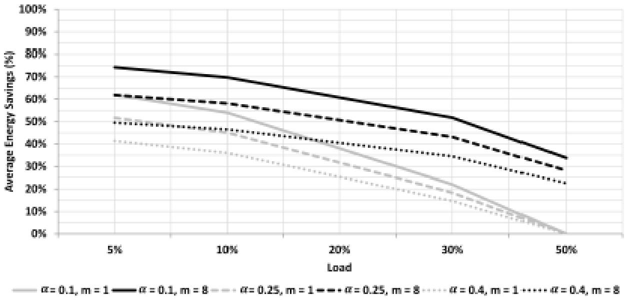


Figure 6: Average energy savings as function of load and different values of α and m.

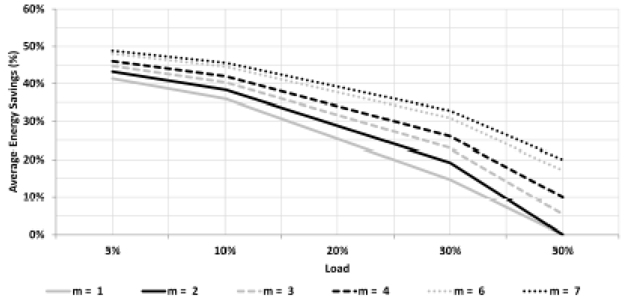


Figure 7: Average energy savings as function of load and different values of m with $\alpha = 0.4$.

bowered. We model the merge trace network and port virtualization in software using parameters from our prototype [7] for reference. For our simulations we use 8 _ 8 merge networks.

In order to model the trace in the network, we rely on several previous studies. The authors in [2] examine the characteristics of the packet-level communications inside different real datacenters including commercial cloud, private enterprise and university campus datacenters. They note that the packet arrivals exhibit an ON/OFF pattern. The distribution of the packet inter-arrival time _ts the Lognormal distribution during the OFF period. However, during the ON period, the distribution varies in deferent data Cen tars due to various types of running applications. For example, MapReduce [15] will display deferent inter-switch trace characteristics than typical university datacenters. Further more, trace between nodes and switches displays patterns quite deferent from the inter-switch trace [3, 16, 17]. Typically, however, the deferent trace patterns _t one of Lognormal, Weibull and Exponential. We can consider the ex-

potential distribution as the most restrictive one among the various indented distributions and we use it to represent the general distribution of the packet inter-arrival times. In order to obtain a comprehensive view of the beets and challenges of using the merge network, we use deferent average trace loads on each link. The values we use are: 5%, 10%, 20%, 30%, and 50% of the maximum link capacity of 40Gbps. The duration of each simulation is 24 hours. In addition, each run is repeated 10 times and the average performance values have been calculated and plotted. The metrics of interest are: energy savings, packet loss

due to merging trace, aggregate throughput achieved and end-to-end (e2et) time. We note that the increased latency due to the merge network is 3 _s (this is based on the time for the selectors in the merge network to sense the presence of packets and appropriately conger the network to switch the packet, please see [7]). 5.1 Results Figure 8 plots the average number of asleep interfaces for the different datacenter topologies and conjurations. The attended buttery topologies shows the best results with a 50% of asleep interfaces with 50% of load. However, the other topologies show poor performance when the load approaches 50%. In particular the 1-D conjuration for the mesh and hypercube topologies have a very small percentage

of asleep interfaces compared to that of 2-D conjurations. In the 2-D case, the hypercube topology has slightly better results than the mesh one. In summary, the attended buttery topology has the best features to achieve significant energy savings due to a greater number of links connected

to other switches m, that can be put in a low power mode.

Conclusions

The paper studies the idea of trace merging in datacenter networks. Earlier work by the author developed the notion of using an analog merge network to aggregate trace from multiple links in an enterprise network and feed it to a much smaller port-density switch. The current paper extends the

idea to datacenter networks where trace merging is shown to enable large number of switch interfaces to operate in low power modes while having no impact on trace. The paper explores the application of trace merging to the attended buttery, mesh and hypercube networks. It is shown that the attended buttery yields almost 20% energy savings even under 50% loading while at 5% load it shows an almost 50% energy savings. The mesh and hypercube networks also show energy savings at all loads but are not as energy efficient as the attended buttery. a theoretical model for energy savings for these networks.

References

[1] D. Bats, M. Marty, P. Wells, P. Klaussner, and H. Liu, \Energy Proportional Datacenter Networks," in Proceedings of the 37th International Symposium on Computer Architecture (ISCA). Saint Milo, France: ACM, June 2010, pp. 338{347.

[2] T. Benson, A. Akella, and D. Maltz, \Network Tra_c Characteristics of Data Centers in the Wild," in Proceedings of the 10th Conference on Internet Measurement (IMC). Melbourne, Australia: ACM, November 2010, pp. 267{280.

[3] B. Heller, S. Seetharaman, P. Mahadevan, Y. Yiakoumis, P. Sharma, S. Banerjee, and Figure 18: End-to-end time as function of load for the (4,2)-ary 2-hypercube and 8-ary 1-hypercube topology.

Figure 19: Average energy savings _es for the 8-ary 2-at attened buttery, (4,2)-ary 2-mesh and (4,2)-ary 2-hypercube topologies. N. McKeown, \ElasticTree: Saving Energy in Data Center Networks," in Proceedings of the 7th USENIX Conference on Networked Systems Design and Implementation (NSDI). San Jose, CA, USA: USENIX Association, April 2010, p. 17.

[4] R. Bolla, F. Davoli, R. Bruschi, K. Christensen, F. Cucchietti, and S. Singh, \The Potential Impact of Green Technologies in Next-Generation Wireline Networks: Is There Room for Energy Saving Optimization? ," IEEE Communications Magazine, vol. 49, no. 8, pp. 80{86, August 2011.

[5] R. Bolla, R. Bruschi, F. Davoli, and F. Cucchietti, \Energy E_ciency in the Future Internet: A Survey of Existing Approaches and Trends in Energy-Aware Fixed Network Infrastructures," IEEE Communications Surveys & Tutorials (COMST), vol. 13, no. 2, pp. 223{244, Second Quarter 2011.

[6] L. Chiaraviglio, M. Mellia, and F. Neri, \Minimizing ISP Network Energy Cost: Formulation and Solutions," IEEE/ACM Transactions on Networking, vol. PP, no. 99, pp. 1{14, 2011.

[7] C. Yiu and S. Singh, \Energy-Conserving Switch Architecture for LANs," in Proceedings of the 47th IEEE International Conference on Communications (ICC). Kyoto, Japan: IEEE Press, June 2011, pp.1{6.

[8] S. Singh and C. Yiu, \Putting the Cart Before the Horse: Merging Tra_c for Energy Conservation," IEEE Communications Magazine, vol. 49, no. 6, pp. 78{82, June 2011.

[9] S. E. Alexander and G. Pugliese. Cordless communications within buildings: Results and measurements at 900mhz and 60ghz. *British Telecom Technology Journal*, 44(10):99 – 105, Oct. 1996.

[10] C. R. Anderson and T. S. Rappaport. In-building wideband partition loss measurements at 2.5 and 60 ghz. *IEEE Trans. on Wireless Communications*, 3(3):922 – 928, May 2004. [3] John C. Bicket. Bit-rate selection in wireless networks. Master’s thesis, MIT, 2005.

[11] J-P. Ebert et. al. Paving the way for gigabit networking. In *Global Communications Newsletter*, April 2005.

[12] G. Fettweis and R. Irmer. Wigwam: system concept development for 1 gbit/s air interface. Wireless OWrld, Rsearch Forum, 2005.

[13] E. Grass, M. Piz, F. Herzel, and R. Kraemer. Draft phy proposal for 60ghz wpan: Ieee p802.15 wg on wpans, November 2005.

[14] Frank Gross. *Smart Antennas for Wireless Communications*. McGraw Hill, 2005.

[15] C. P. Lim, R. J. Burkholder, J. L. Volakis, and R. J. Marhefka. Propagation modeling of indoor wireless communications at 60 ghz. In *IEEE Antennas and Propagation Society International Symposium*, pages 2149 – 2152, 2006.

[16] M. Marcus and B. Pattan. Millimeter wave propagation: spectrum management implications. *IEEE Microwave Magazine*, June 2005.

[17] S. Singh, F. Ziliotto, U. Madhow, E. M. Belding, and M. J. W. Rodwell. Millimeter wave wpan: Cross-layer modeling and multi-hop architecture. In *IEEE INFOCOM (Minisymposium)*, pages 2336 – 2340, May 2007.

[18] Bernard Sklar. *Digital Communications*. Prentice Hall, 2005.