Wireless Technologies in DataCenter Networks

Ziyang Cheng, 5110309523

June 22, 2014

1 ABSTRACT

The datacenter network is the key component of cloud computing. The datacenter network is consisted of thousands of thousands of work stations and switches. However, different traffic demands of different nodes causes the performance of Data center networks (DCNs) worse. Our work aims to solve this problem caused by a few hot nodes to improve the global performance. We model the wireless transmissions in a DCN by considering both the wireless interference and the adaptive transmission rate. We just modulate this problem as an optimization problem and use the genetic algorithm (GA) to address it.

2 INTRODUCTION

Nowadays the development of cloud computing is faster and faster, and more and more data centers are built. The datacenter network is the key component of cloud computing. The datacenter network is consisted of thousands of thousands of work stations and switches. As the infrastructure of data centers, data center networks (DCNs) are constructed to provide a scalable architecture and an adequate network capacity to bear the services.

However, current DCNs come across more and more difficulties with the growth of cloud computing. First, the rapidly increasing size of data centers brings challenges to DCN. For traditional ways, expensive highend switches and a large number of wires are necessary to construct a DCN containing thousands of servers, which leads to great troubles in wiring and maintenance.

On the other hand, data center applications that cause unbalanced traffic distributions suffer from inadequate network capacity. Based on the traffic statistics obtained from a realworld data center, in a traffic demand, there is only a few nodes being hot (i.e., these nodes need to transmit a high volume of traffic). Furthermore, the non-deterministic distribution of hot nodes makes it impossible to set up additional wired links for certain nodes to alleviate their congestions.

To tackle these problems, we propose to utilize wireless transmissions in DCNs. Compared with wired connections, wireless links have advantages in several aspects. First, they are free of wiring and the maintenance is relatively convenient. Second, direct links between servers are easy to achieve with wireless in the scale of a data center, which can avoid the extra cost of multi-hop transmissions. Moreover, variable wireless connections can be set up on-demand. Therefore, it is possible to adjust the topology dynamically to provide more network capacity for hotter nodes. In brief, the flexibility of wireless transmissions provide a feasible approach to address the nondeterministic unbalanced traffic distribution of data center applications.

As a result, a delicate wireless scheduling mechanisms are required to effectively enhance the performance of the whole DCN. For example, wireless links should be established appropriately to alleviate the congestion of hot nodes; channels should be allocated properly to avoid interference.

First, we perform a novel problem formulation for wireless DCN. A realistic interference formalization and the adaptive transmission rate are considered in the model. We use the joint optimization of the throughput of wireless networks and the global job completion time to be the reflection of the performance of the system. Second, we introduce a genetic algorithm (GA) to tackle the channel allocation problem. The GA-based approach can find the solution efficiently, especially when employing inheriting search.

3 RELATED WORK

There has been a lot of research on the interconnection architectures and the routing mechanisms of DCNs. Some of them extend existing tree-based topologies to improve scalability and throughput. Fat-tree [1] groups servers into pods and establishes multiple paths between the core layer and the aggregation layer of a typical tree-based data center architecture. Based on the fat-tree topology, Portland [2] is proposed to support various requirements of data center applications such as virtual machine migration. VL2 [3] is based on Clos Networks, in which new addressing and routing mechanisms are designed to provide high capacity and performance isolation between different services.

Moreover, researchers also try to develop new topologies rather than extend existing ones. DCell [4] takes a structure composed of one switch and k servers as a basic unit and constructs high level topologies recursively by connecting basic units together with direct links between servers. FiConn [5] is an extension of DCell but it only utilizes the backup port of each server rather than add new NICs. BCube [6] introduces more switches to improve the bottleneck problem of DCell and develops a modularized data center solution. It achieves load balancing and a graceful performance degradation under various faulty conditions.

Besides the schemes based on Ethernet, work has also been done to make use of other transmission media. K. Ramachandran et al. first propose to employ 60GHz communications in DCNs [7]. This work designs a clean-slate wireless-based DCN architecture and presents a lot of relevant challenges. However, it does not provide detailed technical approaches. Flyway [8] is the first one that combines wireless networks with existent Ethernet-based DCNs. Yet, it only performs an initial problem formulation and many important factors, including interference and number of radios, are not considered in the scheduling mechanism. Therefore, a lot of problems remain to be investigated to substantiate a wireless DCN. Another work [9] proposes to utilize optical circuit switches for high-speed direct communications between racks. The optical switch is scheduled based on the traffic demands to maximize throughput, which is similar to Flyway.

GA-based approaches have been proposed to handle the channel allocation problem in various wireless networks. Zomaya et al. [10] highlight the potential of using GA to deal with wireless resource allocation and design a GA method with an improved mutation operator to address the problem efficiently. Patra et al. [11] improve the algorithm by introducing a new pluck operator. Ding et al. [12] utilize GA to assign partially overlapping channels in WMN. Our approach is different from the existing ones in that the channel allocation problem in a wireless DCN is different from those in conventional wireless networks (as mentioned in Section I) and we design our own crossover and mutation operators to improve the performance of the GA algorithm.

4 SYSTEM MODEL

4.1 WIRELESS TRANSMISSION

We propose a generic approach to utilize wireless in DCNs such that the adoption of wireless transmissions is independent of the implementation of a DCN. Therefore, the basic unit of a wireless DCN should not be restricted to be a server or a rack. Instead, we formalize it as an abstract concept with the following definition.

Definition 1: A wireless transmission unit refers to a group of servers that uses the same set of antennas to transmit data to other servers outside the group.

Typically, a rack is taken as a unit. For solutions that does not adopt traditional tree-based topologies, we can treat certain specific structures in the corresponding architectures as wireless transmission units.

Based on Definition 1, we classify the traffic in the network into two categories: one is the inter-unit traffic and the other is the intra-unit traffic. Note that wireless links are employed for transmitting the inter-unit traffic. Assume v_1 and v_2 are two units. Let $t(v_1, v_2)$ denote the traffic demand from v_1 to v_2. The distribution of inter-unit traffic can be illustrated with a wireless transmission graph as defined in Definition 2.

Definition 2: A wireless transmission graph is a directed graph $G = (V, E)$, where V denotes the set of units and E denotes the set of transmissions.Each node v in the graph corresponds to a physical unit with antennas. We use $\omega(v)$ to denote the number of antennas belonging to v. An edge $e = (v_1, v_2)$ presents in the graph if and only if the volume of the traffic from v_1 to v_2 is more than 0.

4.2 Channel Allocation and Interference

For a given wireless transmission graph, channels should be assigned to the edges to carry out wireless communications. In this work, we assume different channels are orthogonal. Let C denote the set of channels. When allocating channels, we assign each edge $e \in E$ with an integer $c(e) \in \{0, 1, ..., |C|\}$, in which each non-zero integer corresponds to a certain channel and 0 means assigning no channel to e. Note that not all the wireless transmissions should be carried out simultaneously because some of them may cause serious interference to others and therefore have a negative impact on the global performance. If an edges is assigned a channel, it is called an active edge; otherwise, it is an idle edge.

The set of channels allocated to all the edges form a channel allocation scheme of the wireless transmission graph. Assuming $|E| = n$ (we follow this assumption in the remaining sections of this paper), the channel allocation scheme can be expressed by a vector $X = (x_1, x_2, ..., x_n)$, in which each element x_i stands for the channel assigned to a specific edge e_i.

One of the problems in channel allocation is that the transmission on an edge is possibly interfered by the nearby transmissions working on the same channel.

Definition 3: The conflict edge of an edge e in a wireless transmission graph is the edge whose transmission causes interference on the transmission of e.
The decision of a conflict edge involves the physical position of the nodes and the assigned channels. With regard to physical position, we adopt the interference range model, in which a sender node causes interference on all the nodes within its interference range. Note that our model does not rely on certain antenna techniques. The interference range of a node with an omni-directional antenna is usually defined as a unit disk while that of a directional antenna depends on the relative position of the two endpoints and the beam-forming patterns. Whichever antenna techniques is employed, we just adopt the corresponding interference range model.

Data transmissions in DCNs should be reliable so acknowledgment is required. We transmit data packets and acknowledgment packets at reversed edges. Thus, the transmission on an edge $e = (v_1, v_2)$ is unidirectional, i.e. packets are only sent from v_1 to v_2. Based on the interference range of a node, we can induce the interference range of an edge: An edge $e = (v_1, v_2)$ is in the interference range of another edge $\bar{e} = (\bar{v}_1, \bar{v}_2)$ if v_2 is in the interference range of $\bar{v}_1$.

If e is in the interference range of $\bar{e}$, we consider $\bar{e}$ as a potential conflict edge of e. If $\bar{e}$ is a potential conflict edge of e and $c(\bar{e}) = c(e) \neq 0$, then $\bar{e}$ is the conflict edge of e. Let $\Gamma(e)$ denote the conflict edge set of e and $\Gamma_0(e)$ be the potential conflict edge set.

Since all the nodes are static, potential conflict edge sets can be precomputed for a given wireless transmission graph. The interference relationship can be illustrated with a conflict graph, in which each node denotes a transmission and a directed edge (v_1, v_2) indicates that v_1 potentially interferes v_2.

4.3 SINR and Data Rate

In the research on wireless networks, the protocol interference model and the physical interference model are often used to determine the effect of interference [13]. In the protocol interference model, the transmission of an edge is blocked if one of its conflict edges is active. On the other hand, simultaneous transmissions are admitted in the physical interference model as long as the signal to interference and noise ratio (SINR) at the receiver is larger than a threshold T_SINR. We adopt the latter model in this work. Thus, the transmission on $e = (v_1, v_2)$ is successfully performed if and only if:

$$SINR(e) = \frac{P_s(e)}{\sum_{\bar{e}\in\Gamma(e)} P_I(\bar{e}) + N_0} \geqslant T_{SINR}$$

where $P_S(e)$ denotes the signal power received by v_2, N_0 is the environment noise, and $P_I(\bar{e})$ denotes the interference power caused by $\bar{e}$ and received by v_2. For a given edge e, the edges in $\Gamma(e)$ may cause interference of different intensity on e. We define the intensity of interference as follows.

Definition 4: If $\bar{e}$ is in the potential conflict edge of e, the interference factor between $\bar{e}$ and e is the ratio between the power emitted from the transmitting antenna of $\bar{e}$ and the power received by the receiving antenna of e on the same channel.

The interference factor can be computed according to Friis transmission equation as shown as follows, where $\frac{P_r}{P_t}$ is the ratio of the power received by the receiving antenna P_r and power emitted from the transmitting antenna P_t, G_t and G_r are the antenna gains of the transmitting and receiving antennas, respectively; λ is the wavelength and R is the distance; and the exponent α is typically in the range of 2 to 5 as an estimation to the pass-loss effect.

$$\frac{P_r}{P_t} = G_r G_t \left(\frac{\lambda}{4\pi R}\right)^\alpha$$

For simplicity, we assume that all the antennas have the same gain and the same transmit power. If $\bar{e} = (\bar{v}_1, \bar{v}_2)$ is the conflict edge of e, the power of interference caused by $\bar{e}$ is expressed as follows, where $R(e, \bar{e})$ denotes $R(\bar{v}_1, v_2)$.

$$P_I(\bar{e}, e) = \frac{G_r G_t \lambda^\alpha}{(4\pi)^\alpha} \frac{P_t}{R(e,\bar{e})^\alpha}$$

Let $C_I = \frac{G_r G_t \lambda^\alpha}{(4\pi)^\alpha}$. The interference factor between $\bar{e}$ and e can be expressed as follows:

$I(\bar{e},e) = \frac{C_I}{R(e,\bar{e})^\alpha}$

Similar to the computation of the interference factor, the signal power received by v_2 can also be computed based on the Friis equation. In short, the SINR of e can be computed as follows, where $R(e)$ is equal to $R(v_1, v_2)$.

$SINR(e) = \frac{C_I P_t / R(e)^\alpha}{\Sigma_{\bar{e}\in\Gamma(e)} I(\bar{e},e) P_t + N_0}$

SINR is not only the necessary condition of successful transmissions but also an important factor that influences the data rate of wireless links. This mechanism is based on Shannon theorem, as given as follows, where Capacity is the upper bound of the data rate and B is the channel bandwidth.

$Capacity = B\log_2(1+SINR)$

In this work, we assume the data rate is proportional to the capacity. Assuming all the channels have the same bandwidth B and the rate between the data rate and the capacity is β, the date rate of e can be expressed as follows:

$r(e) = \beta B \log_2(1+SINR(e))$
From above, we can compute the data rate of each transmission as long as the channel allocation scheme and the interference relationships are specified.

5 SCHEDULING MECHANISM

Based on the model of wireless data center networks, we propose a centralized scheduling mechanism for wireless transmissions, in which a central controller periodically gathers the information about traffic demands from all the units as well as schedules wireless links for the inter-unit transmissions. The scheduling consists of two steps: the first step is to construct a wireless transmission graph based on the traffic information; and the second step is to perform channel allocation in the wireless transmission graph. We provide the details of the two steps in this section.

5.1 CONSTRUCTING A WIRELESS TRANSMISSION GRAPH

1. Selecting Transmissions:
   When constructing a wireless transmission graph, the central controller converts the traffic demands to a wireless transmission graph for latter scheduling. Although the converting itself is quite easy, the problem lies in the large number of transmissions. As a well-known NP-hard problem, channel allocation is usually handled by using heuristic algorithms, whose time cost grows with the increase of the number of scheduled objects. For DCNs, the huge number of transmissions leads to an excessively high cost.

Therefore, efforts should be made to decrease the size of the channel allocation problem.

A feasible approach is to select a part of the transmissions to construct the graph rather than involve all the transmissions. As for this approach, the key problem is to determine which transmissions to select. Recall the motivations to introduce wireless transmissions into DCNs. It is the high traffic of sparse hot nodes that causes congestion and put off the completion of a job. Therefore, limited wireless channel resources should be used to serve those nodes. In other words, transmissions belonging to the hot nodes should be selected with a high priority.

Furthermore, as mentioned before, the scheduling is a periodical mechanism which means that the channel allocation scheme will be carried out for a period after each allocating operation. Therefore, if the traffic of a transmission is so low that the corresponding wireless link keeps idle for the most of the period, the transmission should be assigned to wired links rather than occupy wireless channel resources.

Besides, a wireless transmission is restricted by the valid transmission range. As for 60GHz communications, the range is about 10m. For $e = (v_1, v_2)$, if the distance between v_1 and v_2 exceeds the valid transmission range, the corresponding edge should be removed from the graph as it is impossible for the antennas to carry out the corresponding wireless transmission.

2. Weighting Transmissions:
   In conventional wireless scheduling approaches, the total throughput is often taken as the metric of performance. However, it is not the case for our problem. As discussed above, nodes with a higher volume of traffic usually finish their transmissions later due to the limit of bandwidth and consequently, put off the global job completion time. Another example is that some flows are expected to experience much longer delay than others via Ethernet transmission because of the static topology and the routing mechanisms. Under either condition, it is obvious that setting up wireless links for certain transmissions is more profitable even if the corresponding data rate is not as high as that of wired links.

   We formalize this property as the utility of the transmission, which reflects the contribution to the global performance made by transmitting the traffic via wireless links. In a wireless transmission graph, each edge e is associated with a weight $u(e)$ that denotes the utility of the corresponding transmission.

   In this work, we employ the network delay to estimate the utility of a transmission. Intuitively, a transmission with a high network delay, caused by either congestion or a long transmission path, is suitable to be assigned to wireless transmissions. Therefore, the utility should be directly proportional to the network delay. We define the utility as follows, where $d(e)$ is the network delay of e and μ is a positive coefficient. Note that utility is a scalar variable.

$$u(e) = \mu d(e)$$

Generally speaking, the network delay can be estimated based on the traffic distribution and the Ethernet architecture. Yet, this work does not focus on how to perform the estimation. In fact, our channel allocation algorithm does not rely on how utility is computed. As long as each edge of the wireless transmission graph is assigned a weight, our scheduling approach can be applied to generate the corresponding channel allocation scheme.

5.2 ALLOCATING CHANNELS

After constructing the wireless transmission graph, channel can be assigned based on the graph. In this subsection, we first formulate the channel allocation problem and then propose a genetic algorithm to handle the problem.

1. Formulation of the Channel Allocation Problem:
   We formalize channel allocation as an optimization problem and the channel allocation scheme is taken as the variable of the problem. As for the objective of channel allocation, we propose Definition 5 to estimate the impact of a wireless transmission on the global performance based on the definition of utility. The objective function of the optimization problem is the total weighted throughput of all the wireless transmissions.

   Definition 5: The weighted throughput of a transmission is the product of its throughput and its utility.

   Several constraints should be considered in channel allocation. First, the number of active edges belonging to a node should not be more than the number of antennas of that node. Second, the assigned channels should be in the available channel set C. Third, for each active edge, its SINR should be higher than the threshold.

   Let $E_s(v)$ denote the set of edges whose source node is v and $E_d(v)$ be the set of edges whose destination node is v. Based on the above analysis, the channel allocation problem can be expressed as follows. The optimal solution of the problem is the channel allocation scheme that meets the constraints and maximizes the total weighted throughput.

$max \Sigma_{e \in E} u(e) r(e)$

2. Genetic Algorithm:
   In this work, we tackle the channel allocation problem with a GA-based scheduling algorithm. The concept of genetic algorithm is to simulate the process of natural evolution, in which the individuals with higher fitnesses are more likely to survive.

   GA is advantageous in solving the channel allocation problem. First, the delicate design of GA enables it to achieve a better performance in handling NP-hard problems than simple heuristics, such as naive greedy search. Second, the channel assignment problem has inherent local optimization property [12]. An allocation scheme for a subnetwork with less interference locally is more likely to be part of the global allocation scheme because the interference range of wireless transmissions is limited. The property fits well into GA because the selection operator and the crossover operator of GA can reserve optimal local allocation schemes. Third, GA does well in handling the traffic demand evolution. The traffic distribution of a period is strongly correlated to that of the previous period. Therefore, the optimal scheme for the previous period is expected to yield an ideal solution for the current period. The convergence can be accelerated considerably by taking the final generation of previous period as the initial generation of current period. We define this approach as inheriting GA search.

   We adopt the roulette wheel selection as the selection operator, where the selection probability $p_s(X)$ of an individual X in a generation X is calculated based on the follows. The interval [0; 1] is divided into subintervals in such a way that each individual corresponds to a subinterval with the length proportional to its selection probability.

   $p_s(X) = \frac{f(X)}{\Sigma_{\bar{X} \in X} f(\bar{X})}$

   When selection is executed, random numbers ranging from 0 to 1 are generated to select individuals. For each random number, the individual that corresponds to the interval including the random number is selected. Each individual can be selected multiple times. Thus, candidate individuals with lower fitness are more likely to be eliminated. The selection operator is detailed in Figure 5.1.

3. Crossover:
   We adopt the single-point crossover in our algorithm, in which two parent individuals are cut off at the same point and the offsprings are produced by combing different parts of the parent individuals together. In order to speed up convergence, we introduce a greedy heuristic rule, which tends to select the point that can generate offsprings with the highest fitness. Note that not all the offsprings generated by the single-point crossover are feasible solutions. The crosspoint is admissible only if both offsprings are

Input: m individuals $\mathbb{X} = \{X_1, X_2, \cdots, X_m\}$
Output: m selected individuals $\mathbb{Y} = \{Y_1, Y_2, \cdots, Y_m\}$

$\mathbb{Y} \leftarrow \varnothing$
$p_s(X_i) \leftarrow \frac{f(X_i)}{\sum_{X \in \mathbb{X}} f(X)}$, for $i = 1, 2, \cdots, m$
$b_i \leftarrow \sum_{j=1}^{i} p_s(X_j)$, for $i = 0, 1, \cdots, m$
for $j = 1$ to m **do**
  Generate a random number in $[0, 1)$, denoted as δ
  Find i such that $b_{i-1} \leq \delta < b_i$
  $\mathbb{Y} \leftarrow \mathbb{Y} + X_i$
end for
return $\mathbb{Y}$

Figure 5.1: Selection algorithm

feasible solutions. Figure 5.2 details the procedure of crossover. For each pair of parent individuals, it takes O(n) time to find the best crossover point.

Input: two parent individuals X_1, X_2
Output: two offspring individuals Y_1, Y_2

$f_m \leftarrow 0$
$Y_1 \leftarrow 0$, $Y_2 \leftarrow 0$
for $i = 0$ to n **do**
  $(Y_1', Y_2') \leftarrow$ single-point crossover of X_1 and X_2 at i
  if Y_1', Y_2' are feasible **and** $\text{Max}\{f(Y_1'), f(Y_2')\} > f_m$ **then**
    $f_m \leftarrow \text{Max}\{f(Y_1'), f(Y_2')\}$
    $Y_1 \leftarrow Y_1'$, $Y_2 \leftarrow Y_2'$
  end if
end for
return Y_1, Y_2

Figure 5.2: Crossover algorithm

4. Mutation:
   In GA, each generated offspring mutates at a certain probability to turn into a new individual. The mutation usually changes part of the DNAs. In this work, we take the optimal solution in the neighborhood of the original individual as the new individual so that the mutation can encourage the convergence of the iteration.

   The concept of neighborhood is given in Definition 6.

Definition 6: Given a wireless transmission graph $G = (V, E)$, the k-neighborhood ($k \in \{1, 2, ..., n\}$) of a solution scheme X is the set of solutions in which each solution has at most k elements that are unequal to the corresponding elements in X.

Let $N(X, k)$ denote the k-neighborhood of X. Assuming X is optimal in N(X; k), the larger the k, the higher the possibility of X being the global optimal solution; if k = n, X is definitely the global optimal solution. It is obvious that it takes a huge cost to find the optimal solution in a large neighborhood. However, we only need to search in a relatively small neighborhood (typically k = 1 or 2) in mutation. Therefore the time cost is tolerable.

Input: origin individual X; mutation probability p_m; neighborhood size k
Output: new individual Y

```
Y ← X
Generate a random number in [0, 1), denoted as δ
if δ < p_m then
    for all Y' in N(X, k) − X do
        if Y' is feasible and f(Y') > f(Y) then
            Y ← Y'
        end if
    end for
end if
return Y
```

Figure 5.3: Mutation algorithm

Figure 5.3 details the procedure of mutation. We traverse the k-neighborhood of the original individual and find the best one, which takes $O(|C|^k)$ time. Similar to crossover, we should also ensure the feasibility of the new solution in mutation.

5. GA-based scheduling algorithm:
   Based on the problem mapping and the designs of selection, crossover, and mutation, we depict the GA-based scheduling algorithm in Figure 5.4.

   In the algorithm, m feasible schemes are taken as the initial generation. Typically, these schemes can be randomly generated. Taking the final generation of the previous period as the current initial generation is an alternative optimization. For each generation, we first compute the selection probability of each individual in the current generation based on their fitness. After that, selection is executed based on the selection probability to get m new individuals. These selected individual sare randomly paired and crossover is performed over each pair. Each offspring individual experiences the mutation at the probability of p_m. Then, these offspring individuals are taken into the next

Input: m initial individuals $\mathbb{X} = \{X_1, X_2, \cdots, X_m\}$; mutation probability p_m; neighborhood size k; termination threshold l

Output: the optimal solution Y

repeat
  $\mathbb{X}_1 \leftarrow Selection(\mathbb{X})$
  Divide the individuals in $\mathbb{X}_1$ into pairs randomly; denote the set of pairs as $\mathbb{X}_p$
  $\mathbb{X}_2 \leftarrow \{Crossover(X_i, X_j) | (X_i, X_j) \in \mathbb{X}_p\}$
  $\mathbb{X}_3 \leftarrow \{Mutation(X, p_m, k) | X \in \mathbb{X}_2\}$
until No evolution occurs for l generations
$Y \leftarrow \arg\max_{X \in \mathbb{X}} f(X)$
return Y

Figure 5.4: GA-based scheduling algorithm

iteration. The iteration is terminated if no evolution occurs during the last l generations, where a generation is considered evolutionary if the highest fitness of its individuals is higher than that of the previous generation. At last, the individual with the highest fitness in the final generation is taken as the solution.

6 CONCLUSION

We present an exploratory investigation on utilizing wireless networks in DCNs. Different from existing works, we take wireless interference and SINR-based data rate into consideration to build a generic model for wireless DCNs. Besides, we take into account the coordination of the throughput of wireless networks and the global performance. Based on these considerations, we study the channel allocation problem and design a GA-based scheduling algorithm.

7 REFERENCES

[1]M. Al-Fares, A. Loukissas, and A. Vahdat, ąřA scalable, commodity data center network architecture,ąś in SIGCOMM ąŕ08: Proceedings of the ACM SIGCOMM 2008 conference on Data communication. New York, NY, USA: ACM, 2008, pp. 63ÍC74.

[2]R. Niranjan Mysore, A. Pamboris, N. Farrington, N. Huang, P. Miri, S. Radhakrishnan, V. Subramanya, and A. Vahdat, ąřPortland: a scalable fault-tolerant layer 2 data center network fabric,ąś in SIGCOMM ąŕ09: Proceedings of the ACM SIGCOMM 2009 conference on Data communication. New York, NY, USA: ACM, 2009, pp. 39ÍC50.

[3]A. Greenberg, J. R. Hamilton, N. Jain, S. Kandula, C. Kim, P. Lahiri, D. A. Maltz, P. Patel, and S. Sengupta, ąřVl2: a scalable and flexible data center network,ąś in SIGCOMM ąŕ09: Proceedings of the ACM SIGCOMM 2009 conference on Data communication. New York, NY, USA: ACM, 2009, pp. 51ÍC62.

[4]C. Guo, H. Wu, K. Tan, L. Shi, Y. Zhang, and S. Lu, ąřDcell: a scalable and fault-tolerant network structure for data centers,ąś in SIGCOMM ąŕ08: Proceedings of the ACM SIGCOMM 2008 conference on Data communication. New York, NY, USA: ACM, 2008, pp. 75ÍC86.

[5]S. Li, D. Chuanxiong Guo Haitao Wu Kun Tan Yongguang Zhang Lu, ąřFiconn: Using backup port for server interconnection in data centers,ąś in INFOCOM ąŕ09: Proceedings of the 28th IEEE International Conference on Computer Communications, 2009, pp. 2276ÍC2285.

[6]C. Guo, G. Lu, D. Li, H. Wu, X. Zhang, Y. Shi, C. Tian, Y. Zhang, and S. Lu, ąřBCube: A high performance, server-centric network architecture for modular data centers,ąś ACM SIGCOMM Computer Communication Review, vol. 39, no. 4, pp. 63ÍC74, 2009.

[7]K. Ramachandran, R. Kokku, R. Mahindra, and S. Rangarajan, ąř60 GHz data-center networking: Wireless => worry less?ąś NEC Technical Report, 2008.

[8]J. P. S. Kandula and P.Bahl, ąřFlyways to de-congest data center networks,ąś in HotNets 09: the 8th ACM Workshop on Hot Topics in Networks, 2009.

[9]J. P. S. Kandula and P.Bahl, ąřYour data center is a router: The case for reconfigurable optical circuit switched paths,ąś in HotNets 09: the 8th ACM Workshop on Hot Topics in Networks, 2009.

[10]A. Zomaya and M. Wright, ąřObservations on using genetic-algorithms for channel allocation in mobile computing,ąś Parallel and Distributed Systems, IEEE Transactions on, vol. 13, no. 9, pp. 948ÍC962, 2002.

[11]S. Patra, K. Roy, S. Banerjee, and D. Vidyarthi, ąřImproved genetic algorithm for channel allocation with channel borrowing in mobile computing,ąś IEEE Transactions on Mobile Computing, pp. 884ÍC892, 2006.

[12]Y. Ding, Y. Huang, G. Zeng, and L. Xiao, ąřChannel assignment with partially overlapping channels in wireless mesh networks,ąś in WICON ąŕ08: Proceedings of the 4th Annual International Conference on Wireless Internet. ICST, Brussels, Belgium, Belgium: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008, pp. 1ÍC9.

[13]G. Brar, D. M. Blough, and P. Santi, ąřComputationally efficient scheduling with the physical interference model for throughput improvement in wireless mesh networks,ąś in MobiCom ąŕ06: Proceedings of the 12th annual international conference on Mobile computing

and networking. New York, NY, USA: ACM, 2006, pp. 2ÍC13.