Reliable Multipath Transmission in Data Center Network

Tan Chen[1] and Guangwu Hu[2]
[1] College of Computer Science, Beijing University of Technology, Beijing 100124, China
[2] Graduate School at Shenzhen, Tsinghua University, Shenzhen 518055, China
Email: chentan@bjut.edu.cn; Hu.guangwu@sz.tsinghua.edu.cn

Abstract —Packet loss degrades the performance of various cloud services and reliability still plays a critical role in data center network. In this paper, we propose an approach to achieve reliability by strategically spreading traffic onto concurrent multiple paths while FEC is employed. Moreover, to fully develop the potential of path diversity, the rate allocation problem across multiple paths is studied and we propose a heuristic polynomial optimal algorithm to find a rate vector to maximize the mathematical expectation of correctly received packets in one FEC group. Extensive experiments are conducted utilizing packet-level traces from real data center networks and the evaluation results demonstrate clearly that our approach achieves high reliability effectively.

Index Terms—Data center network, reliability, path diversity, rate allocation, FEC

I. Introduction

Over the past several years, a wide variety of cloud services have become focuses of attention [1]-[4]. The typical applications distributed across tens to hundreds of thousands servers in data center network include not only Internet-facing applications, such as web search and social network, but also data-intensive applications, which are represented by MapReduce [5]. The near real-time nature of these applications consisting of large scale distributed work flows in data center leads to each work flow should be completed within its deadline.

Nevertheless, packet loss causing retransmission will increase the propagation delay. For data-intensive application, missing deadline means accuracy of aggregate final result will reduce in Partition/Aggregate design pattern. As well as for interactive application, it will degrade the user experience significantly. Hence, reliability still plays a significant role in maintaining application performance, and furthermore, the design of architecture and transmission technology in data center network.

It is widely agreed that path diversity would offer higher aggregate bandwidth while bring enhancements to reliability. This idea also can be applied over data center network smoothly. Several current research [6]-[9] for Data Center Network (DCN) architecture propose to set up much denser interconnects to maximize the network bisection bandwidth. Consequently, utilizing the plentiful spare capacity throughout the data center, edge-disjoint complete graphs will be created natively and multiple parallel paths will be provided between any pair of servers. Fig. 1 illustrates the samples of path diversity in Bcube, FatTree and VL2, in which node 01 is sender and node09 is receiver. For simplicity, we only mark two paths.

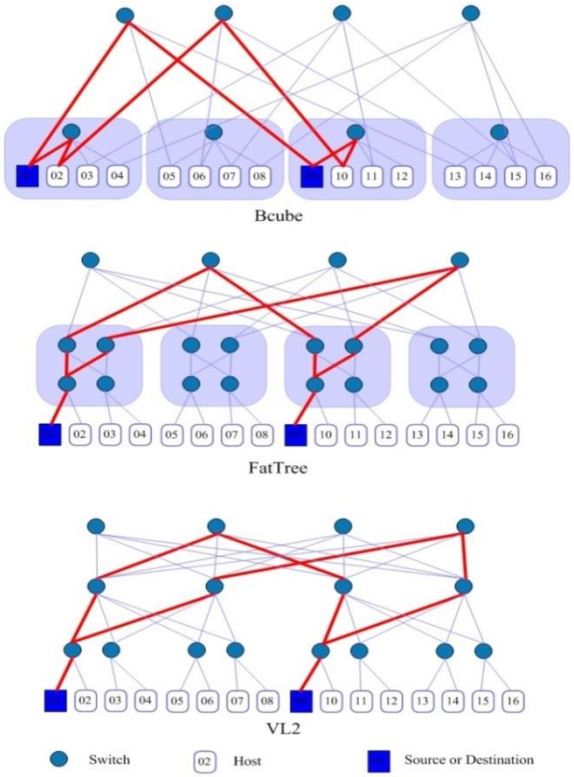

Fig. 1. Path diversity in data center network

Unfortunately, many recent measurement works for data center network [10]-[12] reveal that packet loss occurring within data center network also often exhibits burstiness like Internet, and traffic is generally unpredictable as traffic patterns in data center network changes nearly constantly. Although there are rich connection resources, congestion still may happen when average link utilization is low, moreover, the packet loss is independent of link usage and packet loss probability at links with low utilization may be greater than links with persistently high utilization.

ECMP [13] is a commonly used technique to spread traffic flow across rich multiple redundant paths through

Manuscript received May 6, 2015; revised December 4, 2015.
This work was supported by the China Postdoctoral Science Foundation under Grant No. 2014M560970 and Guangdong Natural Science Foundation under grant No. 2015A030310492.
Corresponding author email: Hu.guangwu@sz.tsinghua.edu.cn.
doi:10.12720/jcm.10.12.939-946

hashing packet's header under the assumption that all paths have equal cost. As well as Valiant Load Balancing (VLB), adopted in the [6], forwarding packets randomly on a per-flow basis, can be viewed as another version of ECMP over a virtual layer-2 infrastructure [14]. However, several researches reveal that ECMP cannot leverage path redundancy in data center network efficiently [11], [12]. Utilizing this static fashion to map traffic flows onto paths can cause collision on some paths, and as a result, although most of links in data center network show low utilization, a small number of links will suffer persistent congestion. Therefore, reliability poses significant challenge, and the improvement of application performance as well as user experience depends in large part on how well the data center network handles this type of bursty loss.

Forward Error Correction (FEC) is a well-known mechanism to combat bursty loss. Through coding M original packets and K redundant packets into one group in sender node, receiver node can decode the whole FEC group successfully and reconstruct original data when it obtains more than M packets even if packet loss event exists. Because FEC mechanism satisfies the delay constraints as well as the introduced bandwidth expansion can be handled well by the aggregate bandwidth, FEC is suitable to be applied in data center network. Hence, the key metric to improve reliability is to develop a technique to ensure receiver node get enough packets.

Let us suppose that packet loss events over each path are irrelevant, the probability of all paths occur occasional congestion and drop packet at the same time is very low. Intuitively, due to existence of consecutive packet loss, if we allocate appropriate traffic for each path according to its loss feature and reschedule packet transmission properly, the existing packet loss pattern will be broken, moreover, this concurrent multipath transmission will perform better than the case only using a single path. Furthermore, combined with FEC, more reliability will be achieved. So far, the problem transfers to how to find the proper optimal rate allocation (RA) over multiple concurrent paths to maximize the expected number of received packets.

The problem of rate allocation across concurrent multipath is shown to be NP-hard [15]. This paper introduces a heuristic optimal algorithm for aforementioned problem in data center network, and we present evaluation results showing that significant reduction in packet loss rate can be achieved by using path diversity together with FEC. It should be noted that our algorithm can also be applied in other network environment, such as Internet and wireless mesh network.

The rest of this paper is organized as follows. Section II describes the system model and the formulation of the problem. In Section III, we present our algorithm to determine the optimal rate allocation. We give the evaluation methodology and results in Section IV and discuss related works in Section V. Finally, Section VI concludes this paper.

II. System Model and Problem Formulation

In this section we discuss the channel model, system model, and the formulation of the problem. Table I summarizes our notations.

TABLE I: Notations

Notation	Refers to
N	Number of paths
M	Number of data packets in a FEC group
K	Number of redundant packets in a FEC group
$S = M + K$	FEC group size
$\boldsymbol{R} = [r_0, r_1, \ldots, r_{N-1}]$	Rate allocation vector
$Q_l(k,s)$	The probability of k out of s packets are lost in path l
$G_l(k,s)$	The probability of k out of s packets are received successfully in path l
$E^{\boldsymbol{R}}_{N,S}$	The expected number of received packets for a FEC block containing S packets over N paths and the rate vector is $\boldsymbol{R}$
$E_{N,S}$	The expected number of received packets for a FEC block containing S packets over N paths
$\tilde{E}_{N,S}$	The optimal expected number of received packets for a FEC block containing S packets over N paths
$\tilde{\boldsymbol{R}}^{\text{opt}}$	The optimal rate vector

A. Channel Model

Since FEC can recover lost packet only if enough packets are received correctly, the bursty loss degrades performance of FEC. Hence to study the reliability of data center network, at first we should understand the behavior of packet loss in DCN. Recently many efforts have been made to capture temporal loss behavior and introduce a variety of Markov models, such as 2-state Gilbert model, the n-state extended Gilbert model, the General Markov model, and the hidden Markov model.

As shown in [16], in comparison with other models, n-state extended Gilbert model, proposed by Sanneck [17], is a more general model for capturing dependencies among loss events, as well as 2-state Gilbert model is a special case of it, meanwhile from the perspective of complexity, n-state extended Gilbert model only requires the past n consecutive loss events, as opposed to remembering n^2 events in the General Markov model. Obviously, a good balance can be achieved between model accuracy and simplicity in n-state extended Gilbert model.

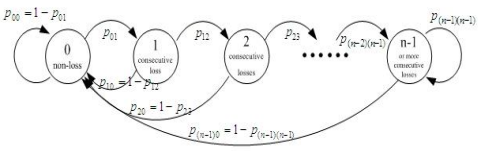

Fig. 2. n-state extended gilbert model

In this paper, we view practicality as a more important factor, hence we assume the n-state extended Gilbert model for end-to-end channel in data center network to model bursty traffic for its simplicity and mathematics tractability. It should be noted that all analytical results in this paper remain valid for any model. Fig. 2 demonstrates how the n-state extended Gilbert model works.

As shown in Fig. 2, a model has n (0,1,…,n-1) states. Each state i indicates there exists exact i consecutive lost packets in current loss event, except for state n-1 which means the run length of consecutive loss is at least n-1 or more. In Markov model, the loss probability vector $\boldsymbol{L}$ = [l_0,l_1,…,l_{n-1}] is another important parameter, while l_i denotes the loss probability of state i. Obviously in extended Gilbert model, $l_0 = 0$ and $\forall i > 0$, $l_i = 1$, therefore the corresponding loss probability vector is $\boldsymbol{L}$= [0,1,…,1] accordingly. The transition probability from state i to state j is denoted by p_{ij}. It is worth mentioning that the n-state Gilbert model assumes only past n consecutive loss events will affect the future. A counter is utilized in the system to remember the number of consecutive lost packets which will return to 0 with the occurrence of a successful transmission. Therefore for a new packet in state i, there are two cases only. It will either be transmitted successfully and reset the counter to 0, or get lost and increase the counter by 1. Hence only two states, $p_{i(i+1)}$ and p_{i0}=1-$p_{i(i+1)}$,are allowed to be successive state for state i, as well as the corresponding transition matrix has only 2n non-zero entries consequently, which is:

$$\boldsymbol{P} = \begin{bmatrix} p_{00} & p_{01} & 0 & \cdots & 0 \\ p_{10} & 0 & p_{12} & \cdots & 0 \\ p_{20} & 0 & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ p_{(n-2)0} & 0 & 0 & p_{10} & p_{(n-2)(n-1)} \\ p_{(n-1)0} & 0 & 0 & p_{10} & p_{(n-1)(n-1)} \end{bmatrix}$$

The steady probability of n-state extended Gilbert model, $\boldsymbol{W}$ = [w_0,w_1,…,w_{n-1}], can be calculated as follows:

$$\boldsymbol{P} \times \boldsymbol{W}' = \boldsymbol{W}', \sum_{i=0}^{n-1} w_i = 1 \quad (1)$$

To calculate its parameters, in this paper we use following equations:

$$p_{01} = (\sum_{i=1}^{n-1} m_i)/m_0 \quad (2)$$

$$p_{(k-1)k} = (\sum_{i=k}^{n-1} m_i)/(\sum_{i=k-1}^{n-1} m_i) \quad (3)$$

where m_i denotes the number of loss events having length i, and k = 2,3,…,n-1.

B. System Model

Fig. 3 illustrates our system model. Forward Error Correction is used for error correction over noisy communication channels. The key idea of this technique is through adding additional redundancy into information before transmission, the destination node can recover all original information even if it receives only a subset of original information in lossy environment. As shown in the figure, in source node, standard FEC code RS(S,M) creates K=S-M redundant packets for original M packets resulting in a total of S packets. Rate allocator computes the rate vector $\boldsymbol{R}$= [r_0,r_1, …, r_{N-1}]($r_i \geqslant 0$, i = 0,1,…,N-1) according to information about performance of each path, and spreads traffic onto N paths. Consequently, S' packets will arrive at the destination node because of the existence of packet loss, where $S' \leqslant S$. Destination node has ability to use correctly received S' to reconstruct all the original packets completely, if and only if $S' \geqslant M$. Obviously, maximizing the mathematical expectation of S' is the optimization objective of rate allocation problem.

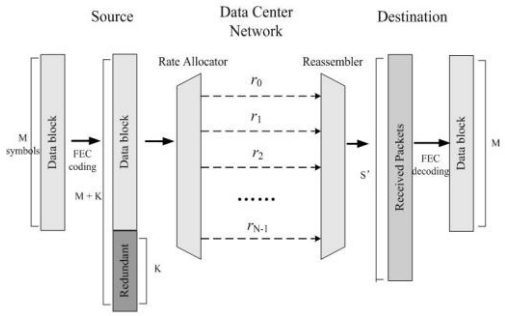

Fig. 3. System model

C. Rate Allocation Problem Formulation

Packet loss degrades the efficiency of FEC if a large number of packets lost and destination node cannot obtain enough packets to recover lost packets. Consider given an FEC group containing S packets, there are $O(N^S)$ways to distribute these packets onto N paths. Due to different loss packet characteristic of each path, different traffic allocation will result in destination node receiving different numbers of packets, in other words, different rate allocation will affect the bursty packet loss behavior. Therefore our objective is to find an optimal rate allocation which can maximize the expected number of received packets to ensure the entire FEC group to be decoded correctly.

The rate allocation problem over multipath in data center network can be formally specified as follows.

Definition: (Optimal Rate Allocation Problem in DCN) Given N independent paths between source node and destination node in data center network as shown in Fig1, given loss feature of each path and a packet block containing S packets which is already performed FEC coding, the objective of optimal rate allocation problems to find a vector $\boldsymbol{R}$ = [r_0,r_1, …, r_{N-1}]($r_i \geqslant 0$, i = 0,1,…,N-1) to maximize the mathematical expectation of received packets, $E_{N,S}$.

$$MaximumE_{N,S} = \sum_{i=0}^{S-1} iPro[recv = i] = \sum_{i=0}^{N-1} \sum_{j=0}^{R_i} j * G_i(j, R_i) \quad (4)$$

Obviously $\sum_{m=0}^{n-1} r_m = S$, and there also exist some other constraints. Let T be the packet block transmit delay constraint required by application, let B_i be bandwidth of path i, the main constraint is $\max_{i=0\ldots N} \frac{r_i}{B_i} \leq T$, otherwise, the whole block will be useless. In this paper we assume the vector satisfies above constraint for simplicity.

III. Rate Allocation Algorithm

RA problem is proved to be NP-hard, and under the assumption of multiple paths are independent, Central-Limit Theorem is always utilized to analyze the RA

problem and be the fundamental basis of solution [18], [19]. However, the study in [15] reveals that Central-Limit Theorem is not the suitable tool.

In this paper, we propose a heuristic polynomial runtime algorithm to compute optimal RA over infinite number of paths from the perspective of practicability. The computation to solve Optimal Rate Allocation Problem in DCN is a process to find the maximum expectation and our basic strategy is decomposing the algorithm into two steps, *E* process and *M* process. In *E* process, we estimate the probability of number of received packet for one path and compute the overall mathematical expectation$E_{N,S}^{R}$ for a given rate vector *R*. In *M* process, a dynamic programming based algorithm is employed to find the rate allocation resulting in the maximum $\tilde{E}_{N,S}$ over multiple paths.

A. E Process

The key point in computing $E_{N,S}^{R}$ is how to estimate probablities of receiving different number of packets. We address this challenge by utilizing *n*-state extended Gilbert model mentioned in Section II (A) to capture the packet loss character.

Let $\boldsymbol{L}$= [l_0, l_1,…l_{n-1}] denote the loss probability vector for *n* states and $\boldsymbol{P}$=[p_{ij}] denote the probability transition matrix. Let $\Gamma_{ij}(k,s)$ denote the probability of the phenomenon in which *k* out of *s* packets are lost while the initial state is *i* and the end state is *j* in one independent path. To compute$\Gamma_{ij}(k,s)$, we partition the problem into two sub-problems, $\Gamma_{iq}(k-1,s-1)$ and $\Gamma_{iq}(k,s-1)$, according to the *n*th packet is lost or not. We solve these sub-problems recursively and then combine solutions. Therefore we can compute $\Gamma_{ij}(k,s)$ as follows:

$$\Gamma_{ij}(k,s)=\sum_{q=0}^{n-1}\big(\Gamma_{iq}(k-1,s-1)l_q p_{qj}+\Gamma_{iq}(k,s-1)\big(1-l_q\big)p_{qj}\big) \quad (5)$$

q is the intermedia state, and obviously, $\Gamma_{ij}(k,s)$ satisfies:

$$\begin{cases}\Gamma_{ij}(k,s)=0 \ \text{ for } k<0 \ or s<0\\ \Gamma_{ij}(0,0)=0 \text{ for } i\neq j\\ \Gamma_{ij}(0,0)=w_i \text{for } i=j\end{cases} \quad (6)$$

where w_i is the steady probability of state *i*. Due to the sub-problems in each step are overlap, we apply dynamic programming algorithm in the computation and utilizing a table to store intermedia solution to avoid redundant computing.

In the case of given $\Gamma_{iq}(k-1,s-1)$ and $\Gamma_{iq}(k,s-1)$, the complexity of computing $\Gamma_{ij}(k,s)$ is O(S^2n), and because we use a table to store intermediate values, the size of memory space we need is *O*(*NSn*).

let $Q_l(k,s)$ denotes the probability of *k* out of *s* packets are lost in path *l* as well as G_l (*k*,*s*) be the probability of *k* out of *s* packets are received successfully in path *l*. We enumerate all of the state transition and accumulate their probability to obtain $Q_l(k,s)$. And G_l (*k*,*s*) can be computed according to $Q_l(k,s)$.

$$Q_l(k,s)=\sum_{i=0}^{n-1}\sum_{j=0}^{n-1}\Gamma_{ij}(k,s) \quad (7)$$

$$G_l(k,s)=Q_l(s-k,s) \quad (8)$$

Therefore, given the rate allocation vector $\boldsymbol{R}$ of a FEC block containing *S* packets, as shown in equation(9), the mathematical expectation of number of successfully received packets on *N* paths can be computed utilizing dynamic programming algorithm.

$$E_{N,S}^{R}=\sum_{i=0}^{N-1}\sum_{j=0}^{r_i} j*G_i(j,r_i) \quad (9)$$

The complexity of computing $G_l(k,s)$ and $E_{N,S}^{R}$ is $O(n^2)$ and *O*(*NS*) respectively, and the overall complexity of this step is $O(NS+n^2+S^2n)$,as well as the total size of memory space we need is $O(NSn+NS^2)$.

B. M Procss

The next task is to find the optimal rate allocation $\boldsymbol{R}$ which result in the maximum expected number of received packets, $\tilde{E}_{N,S}$.The main challenge is the huge search space, there are N^S ways to distribute a FEC block containing *S* packets onto *N* paths.

Algorithm OptRA(*N*, *S*)
1. **for***l*← 1:*N*
2.**for***i* ← 0:*S*
3. **for***p* ← 0:*i*
4.$\overrightarrow{\boldsymbol{G_l^p}}$← {$G_l$ (*0*,*p*), G_l (*1*,*p*), ⋯,G_l (*p*,*p*) }
5.$\overrightarrow{\boldsymbol{r_l^p}}$← {0,1,2, ⋯, *p*}
6.$e_l^p \leftarrow \overrightarrow{\boldsymbol{r_l^p}}\cdot\overrightarrow{\boldsymbol{G_l^p}}$
7. E_l^i ←e_l^p + $\tilde{E}_{l-1}^{i-p}$
8.**if**E_l^i>$\tilde{E}_l^i$
9.$\tilde{E}_l^i$←E_l^i
10.r_l^i← *p*
11. **for***l*← *N*:−1:1
12.$\tilde{r}_l$ ←r_l^s
13.*S* ←*S* − $\tilde{r}_l$
14.$\tilde{\boldsymbol{R}}^{\mathbf{opt}}$ = {$\tilde{r}_0$,$\tilde{r}_1$,$\tilde{r}_2$,…,$\tilde{r}_{N-1}$}
15. **return**$\tilde{\boldsymbol{R}}^{\mathbf{opt}}$

Fig. 4. Algorithm OptRA

Firstly we analyze the structure of $\tilde{E}_{N,S}$, and it is easily to observe that the optimal solution to $\tilde{E}_{N,S}$incorporates the related sub-problems, $\tilde{E}_{N-1,S'}(S'\leq S)$. Hence, our basic strategy is based on *E* process, we compute the value of $\tilde{E}_{N,S}$ recursively in a bottom-up fashion. Furtheremore, because the subproblems are overlaopping, memoization algorithm, an alternative approach to dynamic programming which saving the results of each sub-problem in a table, is applied in our algorithm to achieve efficiency.

The pseudocode of the algorithm is illustrated in Fig.4. The algorithm takes the number of paths *N*and the size of FEC block *S* as inputs, and it returns the rate allocation which results in the optimal expected number of received packets.

Let E_l^i be the mathmatical expect of spread *i* packet s onto *l* paths, $\tilde{E}_l^i$ denotes the optimal E_l^i as well as r_l^i represents the rate for *n*th path to achieve $\tilde{E}_l^i$. Obviously r_l^i is the key factor to compute $\tilde{\boldsymbol{R}}^{\mathbf{opt}}$ and we store each r_l^i in a table for inquiry after obtain it.

The algorithm enumerates all possible allocations for 1 to *N*-1 paths to compute the solution of subproblem. The innermost **for** loop, in lines 3-10, tries each remainder packets for *n*th paths, combined with the optimal solution of subproblem for *n*-1 paths to determine which allocation vector can result in the optimal expected number. $\overrightarrow{\boldsymbol{G}_{\boldsymbol{l}}^{\boldsymbol{p}}}$ denotes the vector of probabilities of receiving from 0 to *p* packets in *l* th path, as well as G_l (*i*, *p*) are computed in the *E* process. Lines 8-10 save better value of E_l^i in each iteration, and the corresponding traffic alloction r_l^i for *n*th path. The **for** loop in lines 11-13 iterates the number of path *N* in reverse order, and in each iteration according to the remainder packet number and path number, it is easy to obtainr_l^s, moreover, the $\widetilde{\boldsymbol{R}}^{\mathbf{opt}}$ can be returnedconsequently.Since we precompute each $G_l(k,s)$ and store them, the complexity for *M* process is $O(NS^2)$, and memory space of size $O(NS)$ is required.

C. Packet Sending Algorithm

This subsection discusses for a sending procedure in the source node, how to distribute a FEC group onto multiple paths utilizing the optimal RA vector $\widetilde{\boldsymbol{R}}^{\mathbf{opt}}$**.** As mentioned above, the sending behavior of the source node will affect the burst loss event and continuous loss event significantly, so the sending procedureshould spread packets evenly on all *N* paths within the rate contraint determined by the upper application.

Algorithm PacketSpread($N, S, \widetilde{\boldsymbol{R}}^{\mathbf{opt}}$**)**
1. **for***l*← 1:*N*
2. **if**$\tilde{r}_l > 0$
3. $p_l = S / \tilde{r}_l$
4. **else**
5. $p_l = \infty$
6. **for***q*← 1:*S*
7. sending packet *q*using path *j* where $p_j = \min(p_0, p_1, ..., p_{N-1})$
8. $p_j = S / \tilde{r}_j$
9. **for***t* ←1:*N and t*≠*j*
10. $p_t = p_t - 1$

Fig. 5. Packet sending algorithm

In this paper, we use following method to spread packets and the pseudocode is illustated in Fig. 5. In the first **for** loop, line 1-5, we assign each path a point value, p_l, according to the proportion of their rate in the total packet number. In fact $p_l$is interval of sending packets for path *l* compared to single path transmission scheme, and also can be viewed as the priority for path *l*in the sendingprocess, lower point value means higher actual priority.The second forloop, line3-7, shows when sending procedure sends a packet, the path with highest priority will be selected. And after sending, p_l resets to the original values for path *l*, as well as other paths raise their priority.

IV. Evaluation

We have performed a series of experiments on a packet level, discrete event network simulator to study the performance of algorithms presented in this paper. This section introduces the method of our evaluation and then discusses the evaluation results.

To evaluate the effectiveness of our algorithm in a more realistic environment, we utilized packet-level traces presented in [11]. The work of [11] studies the network traffic in 10 data centers including not only data centers offering Internet-facing applications, but also data centers used to MapReduce style applications.

Because repeated sequence numbers in TCP header means retransmission of the corresponding packets according to TCP retransmission scheme, in order to characterize the packet loss behavior, we utilized statistics of sequence number in one TCP connection to approximate packet loss rate for the corresponding network path. We selected more than 600 paths for simulation, and Fig. 6 shows the CDF of raw loss rate of the path set, in which the mean loss rate is 0.1118, meanwhile about 70% of these paths' loss rate is lower than 0.1161.

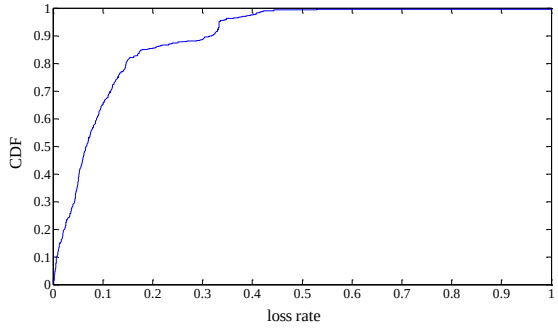

Fig. 6. CDF of paths raw loss rate

We compared the following traffic allocation algorithms in this paper, and the algorithm in each test is applied in the interval of two FEC groups to compute the next allocation.

- Equal distribution (Equal D): Packets are scheduled to multiple paths in equal portions and in circular order, i.e., in a round robin fashion.
- Best path distribution (Best Path D): The procedure always allocates all of packets to the path who has lowest loss rate. In case of paths have same quality, the procedure selects one path randomly.
- OptRA: The algorithm presented in this paper.
- SubRA: Suboptimal rate allocation algorithm introduced in [15].

In our simulation, we utilized *RS* code whose complexity is $O(N^2)$. It should be noted that other erasure codes such as LT code, Raptor code are also can be applied in this mechanism.

For all evaluations in this paper, the size of FEC group is set to be 40 as well as we applied different numbers of redundancy packet, i.e., *K*=4, 8, 12 and 16. In order to compare the performance of different rate allocation schemes, we adopt probability of successfully decoding whole FEC group as the metric of performance. Each experiment was performed 100 times and the mean results are presented.

The simulation was employed 10 times and in each test we randomly selected two paths whose loss rates are in the range of [0.92, 0.97]. Table II depicts the details. The results show that OptRA and SubRA achieve better performance than other two algorithms while Best Path D outperforms Equal D in all cases. When the number of redundant packets is small, the difference in performance of different algorithms is very obvious, as well as with the value of K increases, the gap between these algorithms becomes smaller. This phenomenon indicates that study on rate allocation algorithm is very valuable for the lossy environment where the bandwidth is limited. SubRA achieves similar performance with OptRA, but it needs to be called N times for N paths, instead of obtaining all results just running once. The complexity of SubRA, which is $O(S^2M^2+S^3N+n^2S^2)$, is obviously higher than OptRA. We also observed in simulations that in the case of running long time without re-computing the rate, the Best Path D will dominate all others.

TABLE II: PROBABILITY OF DECODING FEC GROUP SUCCESSFULLY UNDER TWO PATHS

algorithm	K=4	K=8	K=12	K=16
OptRA	0.6697	0.9432	0.9965	0.9999
SubRA	0.6701	0.9426	0.9970	0.9999
Equal D	0.6294	0.9281	0.9913	0.9994
Best Path D	0.6423	0.9374	0.9944	0.9998

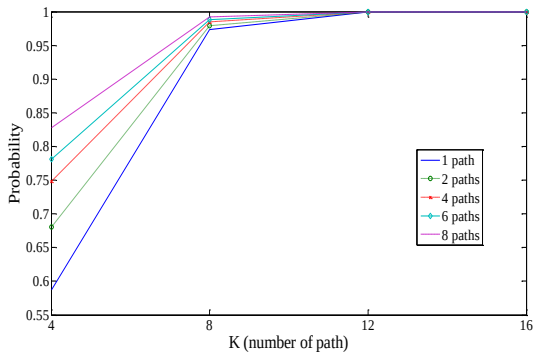

Fig. 7. Rate allocation using 5-states gilbert model

Fig. 7 plots performance comparison between various levels of FEC protection and different numbers of paths whose loss rates are in the range of [0.88,0.92] for 5-state Gilbert Model. In the figure, horizontal axis represents the value of K, the number of redundant packets, as well as vertical axis represents the successful decoding probability. These illustrative results show that as the number of paths and K increase, the successful probability in general increases. For single path transmission, when K=4, the probability is 0.5871. Increasing the number of path to 8, the corresponding probability increases to 0.7477. Furthermore, if we increase K to 8 at the same time, the probability will achieve 0.9919.

We also show the performance for 3-state Gilbert Model in Fig. 8 while the loss rates of paths are in the range of [0.91, 0.94]. Although we used less states to model packet loss, we got the similar overall trend. Curve of single path which does not have path diversity exhibits poor performance, whereas the case of 8 paths consistently exhibits high quality.

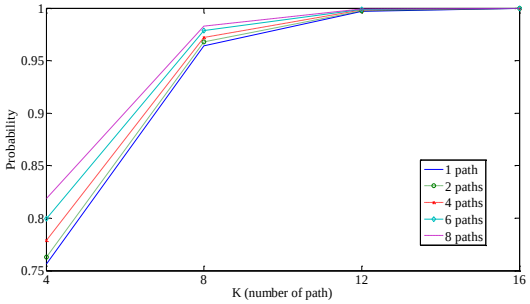

Fig. 8. Rate allocation using 3-state gilbert model

Fig. 9 illustrates the dynamic process of rate allocation of our algorithm, using probability and rate as function of time. In this evaluation we set N=2 , K=8 and use 5-states Gilbert Model. Fig. 9(a) shows results of three cases, using 2 concurrent paths and using path 1 and path 2 along respectively. As shown in the figure, path 1 is better than path 2. From Fig. 9 (b), we can see in order to pursuit high reliability, although our algorithm assigns more than 30 packets onto path 1, it still allocates a small part of packets to path 2.The experimental result in Fig. 9(a) indicates that the path diversity with our algorithm achieves the highest probability.

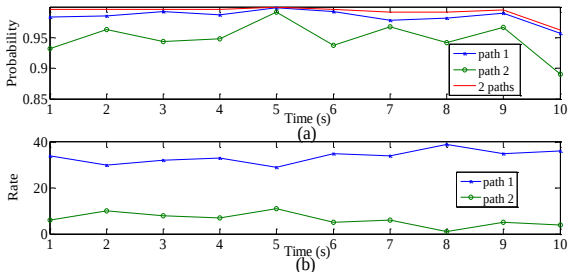

Fig. 9. Rate allocation over 2 paths

V. RELATED WORK

We broadly classify the related work into following two categories: a) path diversity in data center network and b) rate allocation of path diversity in Internet and wireless network.

A. Path Diversity in Data Center Network

Recently, many researches on data center network architecture utilize rich path diversity to provide high bandwidth and low latency for wide varieties of cloud computing services. As well as under the assumption that all paths have equal cost, ECMP is the most commonly adopted scheme in data center network to utilize rich path diversity efficiently through splitting workflows and distributing them onto multiple paths according to a hash of five-tuple in each packet. Similarly, VLB spreads traffic flows randomly onto multiple paths over a virtual layer-2 infrastructure. In order to uncouple the tight tie between routing protocol and specific topology, Ref. [20] proposes GARDEN, an addressing, routing and traffic scheduling protocol on arbitrarily layered topologies for

data center network. GARDEN forms the network utilizing a multi-rooted tree structure and employs the multiple-locator mechanism to exploit path diversity resulting in bringing efficient support for path diversity routing, load balancing and fault tolerance.

In order to leverage topological advantage, Fung Po Tso *et al*. [21] seek to introduce traffic engineering technique into data center network gracefully to fully exploit path diversity. They implement a practical Penalizing Exponential Flow splitting (PEFT) algorithm for DCN and modify link weight optimization, hence routers running PEFT will split and forward traffic for a set of unequal cost paths locally and independently. As a result, they improve network utilization and capacity more efficiently with the performance gain of at least 20 percent over ECMP. Baatdaat [22] is another flow-based scheduler for data center network to exploit topological redundancy and Baatdaat consists of OpenFlow switches. With the help of a single OpenFlow controller to collect link utilization statistics among aggregation switches, Baatdaat schedules flows over multiple paths to reduce maximum link utilization and improve flow completion time.

The work in [23] considers multicast in DCN. In order to take advantage of the rich path diversity commonly available in data center network, they make different multicast groups use different routing trees which leads to more balanced link utilizations and avoids congestion. Moreover, their scheme improves application data rate by up to 12%, and lowers packet loss by51%,on an average in comparison with traditional IP multicast.

The research literatures [24], [25] focus on using muiltipath TCP as a replacement for regular single path TCP for data center network. Raiciu *et al.* [24] propose MPTCP to effectively and seamlessly utilize available bandwidth. MPTCP achieves improved throughput and better fairness on many topologies by exploring multiple paths simultaneously, balancing the load on several TCP sub-flows over different physical path and further moving traffic away from congestion. In [25], an enhanced version of MPTCP, named A-MPTCP, is proposed. A-MPTCP defines and implements a cross-layer cooperation module, enhances its sub-flow creation mechanism so that an adequate number of sub-flows considering the underlying path diversity can be created.

B. Rate Allocation of Path Diversity

Recently, many efforts have been done to address the problem of rate allocation over multiple concurrent paths in wireless network and Internet. Ref. [26] concerns a scenario of multiple senders transmitting packets to a single receiver simultaneously. Assuming that all paths are independent, the work in [26] models path utilizing Gilbert Model and further, with FEC, a brute force search-based receiver-driven algorithm is proposed to spread packets among paths to enhance reliability. The work of Djukic and Valaee [16] focus on improving transmission performance by encoding group of packet using FEC and then transmitting fragments using multiple disjoint paths in wireless network. Based on an in-depth theoretical analysis, they study two rate allocation algorithms, blind load balancing and optimal load balancing, the difference is whether the receiver as path information or not. Their work reveals that even using blind load balancing which distributes traffic among paths uniformly, the lower Probability of Packet Loss (PPL) is achieved than single path transmission in most cases. Similar work is presented in [17]. Li *et al* model network with *M*-states Markov model and they apply Central-Limit Theorem to approximate the distribution of total number of packets among all paths with normal distribution, supposing the environment has a large number of packets distributed over a large number of paths. Based on the distribution, they proposed a pseudo polynomial algorithm to compute optimal rate allocation for all paths in order to achieve minimize expected number of lost data packets utilizing systematic FEC codes. However, the foundation of their algorithm, Central-Limit Theorem and normal distribution, is shown that not the suitable tools to solve the rate allocation problem in [15]. The research work in [15] aims to find the optimal rate allocation to minimize the probability of irrecoverable loss. Authors use Large Deviation Principle (LDP) to compute the distribution of lost packets, and their theoretical analysis proves that the probability of irrecoverable loss decays exponentially with the number of path. Furthermore, they present a heuristic suboptimal algorithm to compute rate allocation for practical environment. However, their algorithm needs to be called *N* times for *N* paths, instead of obtaining all results just running once like ours, i.e., the complexity of their algorithm is higher.

VI. Conclusions

Reliability still plays a critical role in data center network and a natural way to deal with packet loss is utilizing the denser connection resources completely together with FEC. In this paper, we study the rate allocation problem across multiple paths and specify the optimization objective is to maximize the expected number of successfully received packets in one FEC group. Furthermore, a heuristic dynamic programming-based algorithm is proposed to compute the optimal rate allocation in polynomial time. Finally, extensive experiment results show that high reliability can be achieved by using our approach in data center network.

Acknowledgment

This paper was supported by Guangdong Natural Science Foundation (2015A030310492).

References

[1] Hadoop Distributed Filesystem. [Online]. Available: http://hadoop.apache.org

[2] Windows Azure. [Online]. Available: http://www.microsoft.com/azure/
[3] C. Olston, B. Reed, U. Srivastava, R. Kumar, and A. Tomkins, "Pig latin: A not-so-foreign language for data processing," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, Vancouver, BC, Canada, 2008, pp. 1099-1110.
[4] M. Isard, M. Budiu, Y. Yu, A. Birrell, and D. Fetterly, "Dryad: Distributed data-parallel programs from sequential building blocks," in *Proc. 2nd European Conference on Computer Systems, Euro. Sys.*, Lisbon, Portugal, 2007, pp. 59-72.
[5] J. Dean and S. Ghemawat, "Mapreduce: Simplified data processing on large clusters," *Communications of the ACM*, vol. 51, no. 1, pp. 107-113, January 1, 2008.
[6] A. Greenberg, J. R. Hamilton, N. Jain, S. Kandula, C. Kim, *et al*., "VL2: Ascalable and flexible data center network," *Communications of the ACM*, vol. 54, no. 3, pp. 95-104, March 2011.
[7] R. Mysore, A. Pamboris, N. Farrington, N. Huang, P. Miri, *et al*., "Portland: A scalable fault-tolerant layer 2 data center network fabric," *Computer Communication Review*, vol. 39, no. 4, pp. 39-50, 2009.
[8] C. Guo, H. Wu, K. Tan, L. Shi, Y. Zhang, and S. Lu, "Dcell: A scalable and fault-tolerant network structure for data centers," in *Proc. ACM SIGCOMM Conference on Data Communication*, Seattle, WA, United states, 2008, pp. 75-86.
[9] C. Guo, G. Lu, D. Li, H. Wu, X. Zhang, *et al*., "Bcube: A high performance, server-centric network," *Computer Communication Review*, vol. 39, no. 4, pp. 63-74, 2009.
[10] S. Kandula, S. Sengupta, A. Greenberg, P. Patel, and R. Chaiken, "The nature of datacenter traffic: Measurements & analysis," in *Proc. ACM SIGCOMM Internet Meas. Conf. IMC*, Chicago, IL, 2009, pp. 202-208.
[11] T. Benson, A. Akella, and D. Maltz, "Network traffic characteristics of data centers in the wild," in *Proc. ACM SIGCOMM Internet Meas. Conf. IMC*, Melbourne, VIC, Australia, 2009, pp. 202-208.
[12] P. Gill, N. Jain, and N. Nagappan, "Understanding network failures in data centers: Measurement, analysis, and implications," in *Proc. ACM SIGCOMM Conf., SIGCOMM*, Toronto, ON, Canada, 2011, pp. 350-361.
[13] C. Hopps, "Analysis of an equal-cost multi-path algorithm," RFC 2992, IETF, 2000.
[14] B. Theophilus, A. Ashok, A. Aditya, and M. Zhang, "MicroTE: Fine grained traffic engineering for data centers," in *Proc. Conf. Emerg. Networking Exp. Technol., CoNEXT*, Tokyo, Japan, 2011.
[15] S. Fashandi, S. O. Gharan, *et al*., "Path diversity over packet switched networks: Performance analysis and rate allocation," *IEEE/ACM Transactions on Networking*, vol. 18, no. 5, pp. 1373-1386, October 2010.
[16] W. Jiang and H. Schulzrinne, "Modeling of packet loss and delay and their effect on real-time multimedia service quality," in *Proc. 10th International Workshop on Network and Operating System Support for Digital Audio*, June 2000.
[17] H. Sanneck, G. Carle, and R. Koodli, "Framework model for packet loss metrics based on loss run lengths," in *Proc. SPIE Int. Soc. Opt. Eng.*, San Jose, CA, USA, Jan. 2000, pp. 177-187.
[18] P. Djukic and S. Valaee, "Reliable packet transmissions in multipath routed wireless networks," *IEEE Trans. on Mobile Computing*, vol. 5, no. 5, pp. 548-559, September-October 2006
[19] Y. Li, Y. Zhang, *et al*., "Smart tunnel: Achieving reliability in the internet," in *Proc. IEEE INFOCOM*, Anchorage, AK, 2007, pp. 830–838.
[20] Y. Hu, M. Zhu, Y. Xia, K. Chen, and Y. L. Luo, "Garden: Generic addressing and routing for data center networks," in *Proc. IEEE Int. Conf. Cloud Comput., CLOUD*, Honolulu, HI, United States, 2012, pp. 107-114.
[21] F. P. Tso and D. P. Pezaros, "Improving data center network utilization using near-optimal traffic engineering," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 6, pp. 1139-1148, 2013.
[22] F. P. Tso, G. Hamilton, R. Weber, C. S. Perkins, and D. P. Pezaros, "Longer is better: Exploiting path diversityin data center networks," in *Proc Int. Conf. Distrib. Comput. Syst.*, Philadelphia, PA, United States, 2013, pp. 430-439.
[23] A. Iyer, P. Kumar, and V. Mann, "Avalanche: Data center multicast using software defined networking," in *Proc. Int. Conf. Commun. Syst. Networks, COMSNETS*, Bangalore, India, 2014.
[24] C. Raiciu, S. Barre, C. Pluntke, A. Greenhalgh, D. Wischik, and M. Handley, "Improving datacenter performance and robustness with multipath TCP," in *Proc. ACM SIGCOMM Conf., SIGCOMM*, Toronto, ON, Canada, 2011, pp. 266-277.
[25] M. Coudron, S. Secci, G. Pujolle, P. Raad, and P. Gallard, "Cross-Layer cooperation to boost multipath TCP performance in cloud networks," in *Proc. IEEE Int. Conf. Cloud Networking, Cloud Net*, San Francisco, CA, United States, 2013, pp. 58-66.
[26] T. Nguyen and A. Zakhor, "Path diversity with forward error correction (pdf) system for packet switched networks," in *Proc. IEEE INFOCOM*, vol. 1, pp. 663-672, 2003.

Tan Chen was born in 1976. He received his Ph.D. degree in Computer Software and Theory from Beihang University in 2009. He is now a lecturerin the College of Computer Science, Beijing University of Technology. His research interests include data center network and wireless sensor network.

Guangwu Huwas born in 1980, he received the Ph.D. degree in Computer Science and Technology from Tsinghua University. He is now a post-doctor candidate in the Graduate School at Shenzhen, Tsinghua University. His research interests include software defined networking, Next-Generate Internet and Internet security.