An Asymptotically Minimal Node-degree Topology for Load-Balanced Architectures

Zhenhua Liu, Xiaoping Zhang, Youjian Zhao, Hongtao Guan
Department of Computer Science and Technology, Tsinghua University
Tsinghua National Laboratory for Information Science and Technology
Beijing, P. R. China

Abstract—**Load-balanced architectures appear to be a promising way to scale Internet to extra high capacity. However, architectures based on mesh topology have a node-degree of *N*, which prevents these architectures from large node numbers. This consideration motivates us to study the properties of node degree and its impact on the corresponding load-balanced architectures. In this paper we first show the asymptotically minimal node degree for any topology to achieve a constant *ideal* throughput under uniform traffic pattern when the channel bandwidth is fixed. We further introduce a unidirectional direct interconnection topology, named Plus 2^i (P2i), with this minimal node degree and prove that it has an *ideal* throughput of no less than twice the channel bandwidth under uniform traffic pattern. Based on the property, we provide the P2i Load-Balanced (PLB) architecture. Using this architecture, we show that scalability, 100% throughput and packet ordering can be all achieved and the scheduling algorithm is easy to implement. To the best of our knowledge, this is the first load-balanced architecture constructed on multi-hop direct interconnection topologies without packet reordering problem.**

Keywords- load-balanced architecture, direct interconnection networks, P2i, node degree, packet ordering

I. INTRODUCTION

There is an urgent demand to build high-capacity routers with high scalability, throughput guarantees, and no packet reordering. However, traditional single-path routers and multi-path routers later proposed in [1]-[4] suffer from the centralized scheduling and arbitrary per-packet configuration, which prevent them from scaling to fast line rates and high port numbers. These considerations results in a new class of architectures named *load-balanced architectures* [5]-[12]. This class of architectures is based on the idea first proposed by Valiant *et al.* [13], and appears to be a promising way to scale Internet routers to very high capacities [7].

Motivated by the packet reordering in [5], several methods have been proposed to address this problem [6]-[11], which can be further divided into two categories. The first [6]-[8] bounds the amount of packet reordering through the router and depend on a finite reordering buffer at the output. However, when constructing routers of *N* linecards, these approaches need reordering buffers of size $O(N^2)$. The second one is to ensure that packets arrive in order at the output [9]-[12]. However, the methods in [9] and [10] cannot provide (theoretical) 100% throughput guarantee. The approaches in [11] and [12] are based on mesh topology requiring a node degree of N and up to N^2 links which is impractical. The hierarchical mesh architecture in [8] is too complicated to implement. These considerations motivate us to study the properties of node degree and its impacts on the corresponding load-balanced architectures.

In this paper, we explore the Asymptotically Minimal Node-degree Topologies (AMNT) for load-balanced architectures. We first prove that when the channel bandwidth is fixed to a constant (without loss of generality, we use 1 throughout the paper), any topology must have a node degree of $\Omega(\log N)$ to achieve a constant *ideal* throughput under uniform traffic pattern, irrespective of node number, which is essential for load-balance architectures. Then we provide a novel unidirectional direct interconnection topology, Plus 2^i (P2i), whose node degree is $\lceil \log_2 N \rceil$. We prove P2i has an *ideal* throughput of at least 2 under uniform traffic pattern for arbitrary node numbers, so P2i is an AMNT. We further design the P2i Load-Balanced (PLB) Architecture and prove that under the P2i Padded Frame (PPF) scheduling algorithm, PLB can achieve 100% throughput without packet reordering problem. Simulation results show that the PLB architecture has a low average delay and small buffer depth. To the best of our knowledge, this is the first load-balanced architecture constructed on a multi-hop interconnection topology with no packet reordering problem. Mesh is actually one-hop, and it is obviously much more difficult to overcome the packet reordering problem on multi-hop topologies because the hop numbers of different source-destination pairs can be different.

The rest of the paper is organized as follows. In Section II, we study the impacts of node degree on the corresponding load-balanced architectures. In Section III, we introduce the P2i topology and show that it is an AMNT. In Section IV, we design the PLB architecture and the PPF algorithm. We further prove that the PLB architecture can achieve 100% throughput without packet reordering problem. In Section V, we provide simulation results of the PLB architecture. Finally we conclude the paper in Section VI.

II. PROPERTIES OF NODE DEGREE AND ITS IMPACTS

As stated in [14], there are three aspects in the design of an interconnection networks: topology, routing algorithm and flow control. Topology determines the bounds of the

978-1-4244-2324-8/08/$25.00 © 2008 IEEE.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2008 proceedings.

performance. Once topology is chosen, routing algorithm and flow control strive to achieve these bounds.

Note that load-balanced architectures spread the traffic to the middle stage queues and then to the output, where we can consider the traffic uniform, so if topology cannot provide constant *ideal* throughput under uniform traffic pattern, it is impossible to construct load-balance architecture even with constant link speedup. Meanwhile, we favor the topology with minimum node degree, which can reduce the implementation overhead. So our goal is to find out the topology with both the minimum node degree and a constant *ideal* throughput under uniform traffic pattern. First, we will provide the definition of *ideal* throughput under uniform traffic pattern [14].

Definition 1 (ideal throughput under uniform traffic pattern): $\Theta_{\text{ideal}} = b/\gamma_{\max} = 1/\gamma_{\max}$,where b is the channel bandwidth and we assume b=1. $\gamma_{\max}$ is the maximum ratio of the bandwidth demanded by any channel to the arrival rate of input ports.

Formally, we have the following results. The proof is in Appendix I.

Theorem 1: For any topology with node degree fixed to a constant *d*, its *ideal* throughput $\Theta_{\text{ideal}} \sim O(1/\log N)$ under uniform traffic pattern.

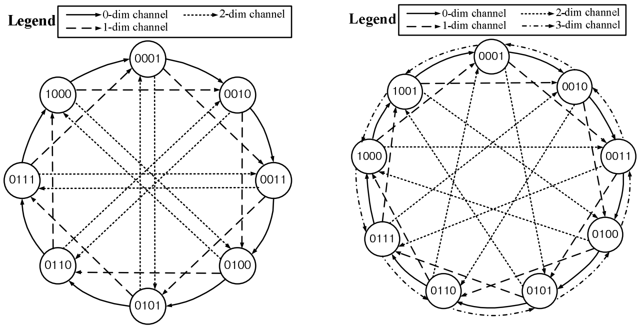
Theorem 1 directly implies the following corollary:

Corollary 1: A topology needs a node degree $d \sim \Omega(\log N)$ to achieve a constant *ideal* throughput under uniform traffic pattern.

III. THE P2I TOPOLOGY

A. Topology Overview

For a P2i topology with N ($2^{n-1}+1 \le N \le 2^n$) nodes, we use $V = \{1, 2, \cdots, N\}$ to denote the node set. For each node *i*, it is connected by n unidirectional channels to nodes $j = (i + 2^k) \bmod N$, $k = 0, 1, \cdots, n-1$. We denote the channel connecting node *i* to $i + 2^k$ by *k-dim* (*k-th* dimension) channel. When $N = 2^n$, we name it *regular* P2i. Otherwise it is named *irregular* P2i. Obviously both *regular* and *irregular* P2i are *symmetric*. Fig. 1 (a) and (b) show examples of P2i topology with 8 nodes (*regular*) and 9 nodes (*irregular*), respectively.



(a) (b)

Figure 1. P2i topology. (a) 8 nodes. (b) 9 nodes.

B. Eigen-Path (EP) routing algorithm

Now we introduce two self-routing algorithms on P2i. The Eigen-Path (EP) routing algorithm routes packets from *i* to *j* as follows: first, transforms $(j-i) \bmod N$ into a binary string, which is called the eigen-vector; second, if the *j-th* highest digit of the eigen-vector is 1, add the corresponding *(j-*1*)-dim* channel to the eigen-path; finally, routes packets in their corresponding eigen-paths from higher dimension to lower one. The algorithm is called Fixed Eigen-Path (FEP) routing algorithm when it forces packets to route by the eigen-paths in fixed order, and is called Arbitrary Eigen-Path (AEP) routing algorithm when the order in the eigen-paths can be arbitrary.

For instance, under the FEP algorithm, in a 16-node P2i, the packet from node 1 to node 12 will route as follows: first $(j-i) \bmod N = (11)_{10} = (1011)_2$; then the eigen-paths are 3-*dim*, 1-*dim* and 0-*dim*; finally route the packets in the following sequence: 1→9→11→12.

C. Properties of P2i

P2i has several merits. Most important of all, it has a node degree of $\lceil \log_2 N \rceil$, which is low enough for extra high linecard numbers (for instance, P2i of node degree 16 can support as many as 65536 linecards), and high enough to ensure a constant *ideal* throughput under uniform traffic pattern when the channel bandwidth is fixed. Formally, we have the following theorem. The proof is in Appendix II.

Theorem 2: With the channel bandwidth 1, the *ideal* throughput of P2i under uniform traffic pattern is

$$\Theta_{\text{ideal}} \ge \Theta_{\text{ideal, R}_{\text{EP}}} = \begin{cases} 2 & \textit{when N is even} \\ 2N/(N-1) & \textit{when N is odd} \end{cases}.$$

where $\Theta_{\text{ideal, R}_{\text{EP}}}$ is the throughput under Eigen Path (EP) routing algorithm, uniform traffic pattern, and perfect flow control.

D. Why Load-balancing?

The FEP algorithm is a satisfactory routing algorithm for uniform traffic. However, when the traffic pattern is not uniform, its performance may considerably deteriorate. For instance, for 32-node P2i, the traffic pattern is as follows: 1 to 32, 9 to 30, 17 to 31, 25 to 29. All of the traffic will travel through the channel from 25 to 29 under the FEP algorithm, so only one quarter traffic can be injected into the input ports compared to that under uniform traffic pattern. Formally, we have the following generalized result, which indicates the necessity of load-balancing. The proof is in Appendix III. The result of the AEP algorithm is similar.

Theorem 3: For worst-case admissible traffic and FEP algorithm, the ideal throughput can be as less as $1/2^{\lfloor n/2 \rfloor}$ of that under uniform traffic pattern.

IV. THE PLB ARCHITECTURE

A. Overview

The PLB architecture is constructed on the P2i topology. As depicted in Fig. 2, every node in P2i corresponds to a nodecard with an input link and an output link in the PLB architecture. We name it nodecard because it behaves as a linecard when switching and as a node when routing inside the architecture. Nodecards are connected by inner links (*0-dim* to (*n*-1)-*dim*) as in P2i. Note that the PLB architecture is *symmetric* as P2i. This architecture is essentially different from the traditional *load-balanced architectures* because it is based on a multi-hop unidirectional direct interconnection topology.

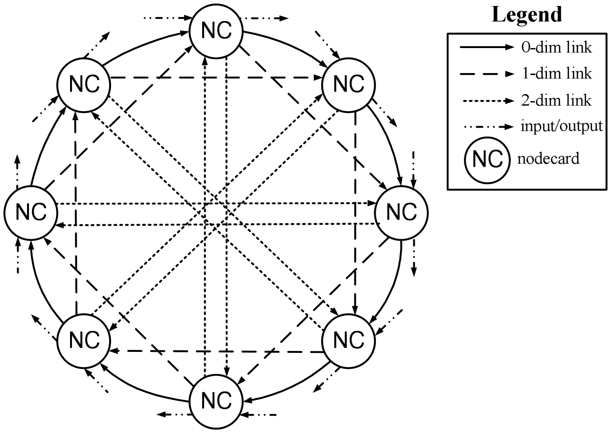


Figure 2. Overview of the PLB architecture (8 nodecards).

Fig. 3 illustrates the block diagram of one nodecard in the PLB architecture. As the PLB architecture is *symmetric*, this is enough to understand the whole architecture.

A nodecard can be divided into the four components: the input block containing the input I-VOQs (VOQ with insertion) [15]; the switching block containing one $n \times n$ crossbar for switching; the intermediate block with the intermediate I-VOQs and the output block. All the four blocks will work according to the PPF algorithm presented in the following part. There are also two arbitrators to decide whether the packets are put in the intermediate I-VOQs (output) or go on routing.

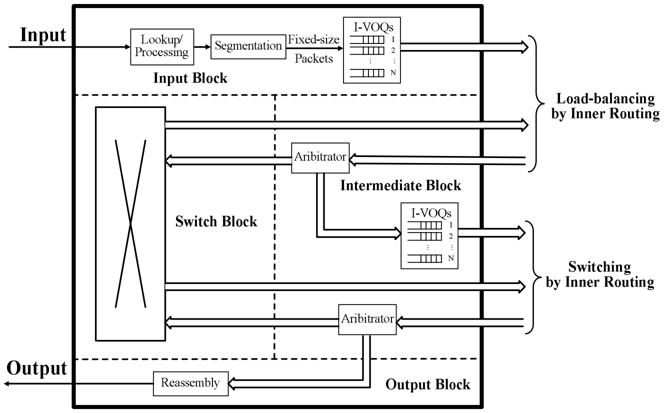


Figure 3. Nodecard block diagram

B. *The Padded Frame in P2i (PPF) scheduling algorithm*

Among the current scheduling algorithms in load-balanced architectures, UPS can maintain packet ordering and achieve 100% throughput but its average delay under light load is considerably high. FOFF improves the average delay at the cost of bounded packet reordering inside the switch, which is eliminated by CMS [11] and PF [12]. The average delay of PF and CMS are lower than FOFF.

Because none of these algorithms have been used on multi-hop direct interconnection topologies, we propose the P2i Padded Frame (PPF) scheduling algorithm combining the idea of PF and the AEP routing algorithm described in III.B.

The PPF algorithm runs independently on each nodecard using information local available. The input nodecard keeps *n* I-VOQs and the incoming packets are placed at the right place decided by the P2i Matching (PM) algorithm. The basic idea is similar to the PF algorithm except that in the PLB architecture, we depend on the PM algorithm to ensure that the picked frames of packets at the input nodecard can be spread into the intermediate nodecards in $\lfloor N/2 \rfloor$ time slots and further forwarded to the right output linecard in another $\lfloor N/2 \rfloor$ slots.

More precisely, the PPF algorithm operates as follows:

1. An arriving packet to input *i* destined to output *j* is placed in input I-VOQ$_{ij}$ according to the PM algorithm.
2. Every *N* time slots, the input select an input I-VOQ to serve for the next *N* time slots. First, it picks round-robin among the input I-VOQs containing at least *N* packets (one full frame). If there are no such I-VOQs, the algorithm searches among the nonempty queues looking for the largest one. Because P2i is *symmetric*, we only consider the input *i*. Assume that the input I-VOQ$_{ik}$ is the largest queue. Let L_k be the length of the *k*-th intermediate I-VOQ of intermediate input *i*, which is on the same nodecard (local information). If $L_k < T$ (*T* is the threshold parameter) then schedule input I-VOQ$_{ik}$ with padded packets, else do not serve input I-VOQ$_{ik}$.
3. In the first $\lfloor N/2 \rfloor$ time slots, spread packets from all inputs to intermediate inputs (load-balancing by inner routing). This is ensured by the property of the PM algorithm stated in the following part. This is the load balancing stage.
4. In the second $\lfloor N/2 \rfloor$ time slots, forward packets from all intermediate inputs (the first packet at every I-VOQ) to outputs (switching by inner routing). This is also ensured by the property of the PM algorithm stated in the following part. This step can be considered as the converse process of step 3. This is the switching stage.

C. *The P2i Matching (PM) algorithm*

The result of the PM algorithm for *N* nodes is a matrix, $\mathrm{M}^N = \left[m_{ij}^N \right]$ ($1 \le i \le N, 1 \le j \le n$), where m_{ij}^N denotes the forwarding sequence of the *j-dim* eigen-path of the packets from source nodecard *s* to the destination nodecard *t* where their eigen-vector $(t-s) \bmod N = i-1$ and $m_{ij}^N = 0$ if the eigen-paths do not contain *j-dim* eigen-path.

The idea of the PM algorithm is as follows: if packets belong to the first *e* lines in M^N can be spread in no more than

$\lfloor e/2 \rfloor$ time slots, then we first spread the packets belong to the first 2^k lines and the *(k+1)-dim* eigen-paths belongs to the following $N-2^k$ lines (at most 2^k eigen-paths and in reverse order) in the first 2^{k-1} time slots; the eigen-paths of the packets belong to the following $N-2^k$ lines can be divided into two parts: eigen-paths lower than (*k*+1)-*dim* and eigen-paths of (*k*+1)-*dim*. For the first parts, these can be spread out in the no more than $\lfloor (N-2^k)/2 \rfloor$ time slots. For the latter parts, if $N \le 2^k + 2^{k-1}$, these (*k*+1)-*dim* eigen-paths have been served in the first 2^{k-1} time slots. Otherwise there are $N-2^k-2^{k-1}$ (*k*+1)-*dim* from line 2^k+1 to $N-2^{k-1}$. Because $N-2^k-2^{k-1} \le \left\lfloor (N-2^k)/2 \right\rfloor$, these eigen-paths can be served in the $\lfloor (N-2^k)/2 \rfloor$ time slots.

Based on the intuitions, we design the PM algorithm to iteratively obtain the matching results for the PLB architecture with *N* ($2^k+1 \le N \le 2^{k+1}$) nodes based on the matching results of PLB with no more than 2^k nodes. The iteratively construction process is in Appendix V. Because the PM algorithm is deterministic, we can ensure the sequence of time slot for packets from source nodecard *s* to the destination nodecard *t* is fixed as long as *N* is fixed. Formally, we have the following property of PM. The proof is in Appendix V.

Theorem 4: When adopting PM, we can uniformly spread one frame (*N* packets) at every nodecard (N^2 packets totally) to every nodecard, one packet each, in $\lfloor N/2 \rfloor$ time slots.

D. Stability of the PLB Architecture

From Theorem 4, we can treat the P2i topology with the PM algorithm as the two crossbars in traditional load-balanced architecture. Therefore, all stable frame-based scheduling algorithms (e.g. PF, UPS, and FOFF) can be adopted to achieve 100% throughput. Formally, we have the following theorems. The proofs are exactly the same with that of the stability of corresponding frame-based algorithms.

Theorem 5 (Stability): The PLB architecture with any stable frame-based algorithm and PM (for instance, PPF) has the same throughput as an ideal output-queued switch, irrespective of the arrival process.

Theorem 6 (Stability with Speedup): The PLB architecture with any stable frame-based algorithm and PM is stable with link speedup *S*.

E. Properties of the PLB Architecture with PPF

In summary, the PLB architecture with the PPF algorithm has the following properties:

1. No pathological traffic patterns. The PLB architecture with PPF has the same throughput as an ideal output-queued switch, irrespective of the arrival process.
2. Packets leave the architecture in order without reordering buffer. Because the PM algorithm is deterministic, we can ensure the sequence of time slot for packets from source nodecard *s* to the destination nodecard *t* is fixed as long as *N* is fixed. So we can rely on the insertion ability of I-VOQs proposed in [15] to insert packets according to the eigen-vector of the intermediate nodecard and destination nodecard to ensure packets leave the architecture in order.
3. All the algorithms are practical to implement. The computational complexity of PPF is the same with that of PF expect for the PM algorithm. Because PF has *O*(1) complexity and PM does not need online computation, PPF has *O*(1) computational complexity.

V. Simulation Results

In this section, we present the simulation results. In all the simulations, we use the Bernoulli traffic model, which is further divided into three kinds: uniform Bernoulli, $\lambda_{ij} = 1/N, \forall i, \forall j$; Tornado Bernoulli, $\lambda_{ij} = 1$ for $j = i + \lceil k/2 \rceil - 1$ and $\lambda_{ij} = 0$ for others; hybrid Tornado Bernoulli, $\lambda_{ij} = h$ for $j = i + \lceil k/2 \rceil - 1$ and $\lambda_{ij} = (1-h)/(N-1)$ for others. Fig. 4 illustrates the average delay under different traffic patterns and loads when *N*=64. From the figure we can discover that the uniform Bernoulli has the highest average delay because it generates the most fake packets. Fig. 5 shows the buffer depth of the intermediate VOQs under hybrid Tornado traffic (*h*=0.6) when *N*=64. We can find that the buffer-depth is quite small under light loads and becomes larger under heavy loads. Because the delay performance greatly depends on the scheduling algorithms, other results are similar to that in [12].

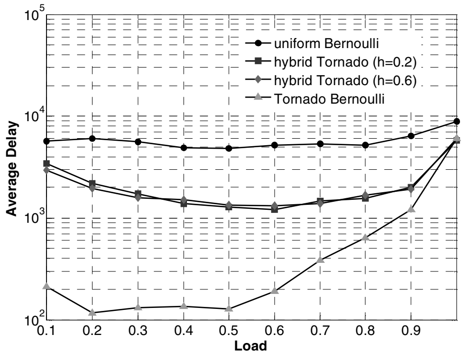


Figure 4. Average delay under different traffic patterns (*N*=64).

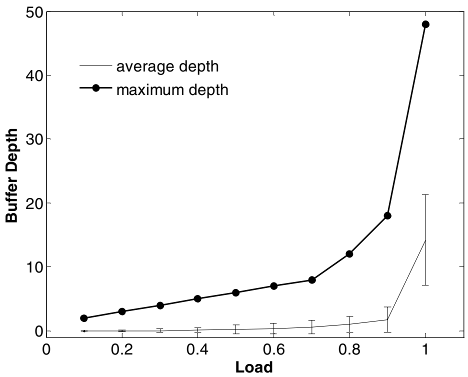


Figure 5. Buffer depth under hybrid Tornado Bernoulli traffic pattern (*h*=0.6, *N*=64).

VI. Conclusions

In this paper we study the problem of the asymptotically minimal node degree for a topology to achieve a constant *ideal* throughput under uniform traffic pattern when the channel bandwidth is fixed. We prove that this asymptotically minimal node degree is $\Omega(\log N)$. Then we introduce P2i with this minimal node degree and prove that it has an *ideal* throughput of no less than 2 under uniform traffic pattern. Based on this property, the PLB architecture is built and is shown to have scalability, 100% throughput and packet ordering and the scheduling algorithm has an *O(*1*)* computational complexity. To the best of our knowledge, this is the first time to build load-balanced architectures on multi-hop direct interconnection topologies without packet reordering problem. In the future we will further study the average delay property of load-balanced architecture based on direct interconnection networks.

Acknowledgment

This research is supported by National Natural Science Foundation of China under Grant No. 90604029, 60773150 and National High Technology Research and Development Program of China under Grant No. 2007AA01Z219.

References

[1] S. Iyer and N. McKeown, "Analysis of the parallel packet switch architecture", IEEE/ACM Transactions on Networking, vol. 11(2), pp. 314–324, April 2003.

[2] S. Iyer, R. Zhang and N. McKeown, "Routers with a single stage of buffering", ACM SIGCOMM '02, Pittsburgh, USA, August 2002.

[3] E. Oki, J. Zhigang, R. Rojas-Cessa and H. J. Chao, "Concurrent round-robin-based dispatching schemes for Clos-network switches," IEEE/ACM Transactions on Networking, vol. 10(6), pp. 830–844, December 2002.

[4] W. J. Dally, P. Carvey and L. Dennison, "Architecture of the Avici terabit switch/router", Proceedings of Hot Interconnects VI, pp. 41-50, August 1998.

[5] C. S. Chang, D. S. Lee, Y. S. Jou, "Load balanced Birkhoff-von Neumann switches, Part I: one-stage buffering," Computer Communications, vol. 25, pp. 611-622, 2002.

[6] C. S. Chang, D. S. Lee, C. M. Lien, "Load balanced Birkhoff-von Neumann switches, Part II: multi-stage buffering," Computer Communications, vol. 25, pp. 623-634, 2002.

[7] I. Keslassy, S. T. Chuang, K. Yu, D. Miller, M. Horowitz, O. Solgaard, and N. McKeown, "Scaling Internet routers using optics," Proceedings of ACM SIGCOMM, Karlsruhe, Germany, 2003.

[8] I. Keslassy, "The Load-Balanced Router," Ph.D. Thesis, Stanford University, 2004.

[9] C. S. Chang, D. S. Lee, Y. J. Shih, "Mailbox switch: a scalable two-stage switch architecture for conflict resolution of ordered packets," Proceedings of IEEE INFOCOM, Miami, FL, 2004.

[10] Hyoung-Il Lee, A two-stage switch with load balancing scheme maintaining packet sequence, IEEE Communications Letters, 2006, 10(4): 290-292.

[11] B. Lin, I. Keslassy, "The concurrent matching switch architecture", Proceedings of IEEE INFOCOM, Barcelona, Spain, 2006.

[12] J. J. Jaramillo, F. Milan and R. Srikant, Padded frames: a novel algorithm for stable scheduling in load-balanced switches, Proceedings of CISS, Princeton, NJ, March 2006.

[13] L. Valiant and G. Brebner, "Universal schemes for parallel communication", Proceedings of the 13th Annual Symposium on Theory of Computing, pp. 263–277, May 1981

[14] William J. Dally and Brian Towles. Principles and practices of interconnection networks, pp. 51-55, Morgan Kaufmann, San Francisco, CA, 2004

[15] C. L. Yu, C. S. Chang and D. S. Lee, CR switch: a load-balanced switch with contention and reservation, Proceedings of IEEE INFOCOM, Anchorage, AK, 2007.

Appendix I: Proof of Theorem 1

For an topology of fixed node degree *d*, its diameter *D* must satisfy the following inequation:

$$1 + d + d^2 + \cdots + d^D \ge N \qquad (1)$$

This is because in this topology, a certain node can have at most *d* nodes with distance 1, d^2 nodes with distance 2, d^3 nodes with distance 3, and so on. The total number of these nodes with distance no more than *D* should be larger than *N*.

(1) implies $d^{D+1} \ge N(d-1)+1$, so we have $D \sim \Omega(\log N)$. Then the average distance $\bar{D}$ can be analyzed as follows:

$$\bar{D} = \frac{1}{N}\sum_{i=0}^{D} in_i \ge \frac{1}{N}\sum_{i=0}^{D-1} id^i \sim \frac{1}{N}\Omega(N\log N) \sim \Omega(\log N) \qquad (2)$$

, where n_i is the number of nodes with distance *i*.

Since all traffic has to be routed, the total bandwidth demand must equal to *N* times the average distance, as shown in equation:

$$\gamma_{\max} Nd \ge N\bar{D} \qquad (3)$$

Combining (2) and (3), we have $\gamma_{\max} \ge D/d \sim \Omega(\log N)$. So $\Theta_{\text{ideal}} \sim O(1/\log N)$. ■

Appendix II: Proof of Theorem 2

In order to prove Theorem 2, we will first prove the following lemma by recursion.

Lemma 1: the bandwidth demand of higher dimension channels is no more than that of lower ones.

Proof: Note that the demand bandwidth of *j-th* dimension channels equals to the sum of the *j-th* digit of every eigen-vector, and further equals to *N* times the sum of the *j-th* digit of the binary numbers from 1 to *N*-1 because P2i is always *symmetric*.

Suppose Lemma 1 holds for $N \le 2^k$, then for $N \le 2^{k+1}$, we can divide the binary numbers from 1 to *N*-1 into two parts: numbers from 1 to $2^k - 1$ and numbers from 2^k to *N*. For the former part, it is obvious that the sum of the *j-th* digit equals to 2^{k-1} when $j \le k$ and 0 when $j > k+1$. For the latter part, because Lemma 1 is true for $N \le 2^k$ and this part meets this condition, the sum of higher digit is no more than that of the lower one except for the (*k*+1)*-th* digit. So Lemma 1 is also true for the sum of digit lower than (*k*+1)*-th* dimension. If $N \le 2^k + 2^{k-1}$, then the sum of the (*k*+1)*-th* digit is no more than that of the *k-th* digit. For the number from $2^k + 2^{k-1} + 1$

to $N-1$, the sum of the (k+1)-*th* digit equals to that of the *k-th* digit. So for $N \le 2^{k+1}$, the sum of the (k+1)-*th* digit is no more than that of the *k-th* digit.

Lemma 1 is obviously true when $k \le 2$, which completes the proof of Lemma 1.

With Lemma 1, we only need to consider the *0-dim* channels. It is not hard to obtain:

$$\Theta_{\text{ideal, R}_{\text{EP}}} = \begin{cases} 2 & \textit{when N is even} \\ 2N/(N-1) & \textit{when N is odd} \end{cases} \tag{4}$$

Combining (4) with the fact that $\Theta_{\text{ideal}} \ge \Theta_{\text{ideal, R}_{\text{EP}}}$, we finish the proof of Theorem 2. ■

Appendix III: Proof of Theorem 3

For any admissible traffic pattern, we have $\sum_{i=1}^{N} \lambda_{ij} \le 1$ and $\sum_{j=1}^{N} \lambda_{ij} \le 1$, where λ_{ij} denotes the fraction of traffic from source *i* to destination *j*. For a *k-dim* channel, the number of source nodes that can send traffic through this channel is 2^{n-k} and the number of destination nodes that can receive traffic through this channel is 2^k. So we have:

$$\begin{aligned} \gamma_{k,\max} &= \max_k \gamma_{k,\max} = \max_k \{\textstyle\sum_{i \in S, j \in D} \lambda_{ij}\} \\ &\le \max_k \{\min\{|S|,|D|\}\} = \max_k \{\min\{2^{n-k}, 2^k\}\} = 2^{\lfloor n/2 \rfloor} \end{aligned}.$$

From the definition of ideal throughput, we can obtain the result of Theorem 3. ■

Appendix IV: The PM Algorithm

```
P2i Matching Algorithm /*Calculate $M^N$ iteratively */
Step 1: Initialization
    $M^2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$
Step 2: Iteration (We already have $M^{2^k}$ )
    for $i \leftarrow 1$ to $2^k$ do
        for $j \leftarrow 1$ to $k$ do
            if $m_{ij}^{2^k} \ne 0$
                then $m_{ij}^N \leftarrow m_{ij}^{2^k}$
            else $m_{ij}^N \leftarrow 0$
        end for
        $m_{i,k+1}^N \leftarrow 0$
    end for
    for $i \leftarrow 2^k+1$ to $N$ do
        for $j \leftarrow 1$ to $k$ do
            if $m_{ij}^{N-2^k} \ne 0$
                then $m_{ij}^N \leftarrow m_{ij}^{N-2^k} + 2^{k-1}$
            else $m_{ij}^N \leftarrow 0$
        end for
        if $N \le 2^k + 2^{k-1}$
            then $m_{i,k+1}^N \leftarrow N-i+1$
        else
            if $i \ge N - 2^{k-1} + 1$
                then $m_{i,k+1}^N \leftarrow N-i+1$
            else
                $m_{i,k+1}^N \leftarrow 2^{k-1} + i$
    end for
Step 3: Output $M^N$
```

Appendix V: Proof of Theorem 4

Lemma 2: The matrix output by the PM algorithm has the following properties:

(a) Every eigen-path of every source-destination pair is assigned to a time slot;

(b) At every time slot, for every dimension, there is at most one eigen-path assigned.

(c) At every time slot, for every source-destination pair, there is at most one eigen-path assigned.

Proof: (a) and (b) is obvious according to the process of PM. We now prove (c) by recursion.

Suppose Lemma 2(c) holds for $N \le 2^k$ and we need to prove that it still holds for $N \le 2^{k+1}$. This is equivalent to prove the eigen-path of the (k+1)-*dim* does not conflict with the eigen path lower than (k+1)-*dim* for source-destination pairs from 2^k+1 to $N-2^k-2^{k-1}$ when $N > 2^k + 2^{k+1}$. In order to prove this, we need another property:

(d) Source-destination pair of lines from 2^i+1 to 2^{i+1} ($1 \le i \le k-2$) can be served in 2^i time slots. Source-destination pair of line 1 does not consume any time slot and source-destination pair of line 2 consumes one time slot.

We now prove that (c) and (d) hold for $N \le 2^{k+1}$ when they hold for $N \le 2^k$. First, note that (d) holds for $N \le 2^{k+1}$ if (a)-(c) hold for $N \le 2^k$. Second, from (d) for $N \le 2^k$ we can obtain that the eigen-paths lower than (k+1)-*dim* for source-destination pairs from 2^k to 2^k+2^i can be all served no later than time slot 2^k+2^{i-1} (2^k for $i=0$) and the time slot for eigen-paths of the (k+1)-*dim* of source-destination pairs from 2^k to 2^k+2^i is later than 2^k+2^{i-1}. So (c) holds for $N \le 2^{k+1}$.

This Lemma is obviously true when $k \le 2$, which completes the proof of Lemma 2.

Lemma 2(a) ensures that every eigen-path of every packet is served, and Lemma 2(b) and 2(c) ensure that there is no contention in the switch (crossbar) of the nodecard. Combining the symmetry of the PLB architecture, we can obtain Theorem 4. ■

978-1-4244-2324-8/08/$25.00 © 2008 IEEE.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2008 proceedings.