

Performance Comparison of Enhanced Data Vortex Networks with Node Buffers and with Inter-cylinder Paths

Qimin Yang

Engineering Department

Harvey Mudd College

Claremont, USA

e-mail: qimin_yang@hmc.edu

Abstract— Optical switching fabric networks have become essential systems in high capacity communication and computing systems. This paper focuses on Data Vortex network architecture with two alternative implementations for improved performance. Either a buffer is added within the routing node or inter-cylinder paths are provided for enhanced routing performance. Since the extra hardware required for both implementations are the same, the network with better routing performance provides a better solution. A comparative study of the two methods is conducted with various load conditions and network redundancy. In addition to random traffic, performances under bursty traffic are also studied. The results have demonstrated that networks with inter-cylinder paths provide significantly lower latency and better throughput, and they are especially advantageous under bursty traffics. All results have shown that the approach with inter-cylinder paths provides more effective sharing of the routing resource within the network compared with the node buffering method. The difference in performance is also shown to be more dramatic under higher load conditions and for larger networks. Finally the comparison is also extended to a modified 4-ary Data Vortex network, where traffic backpressure increasingly becomes a limiting factor due to deflection. Under medium to low redundant conditions, a similar performance trend is observed as that in regular binary Data Vortex network, where the inter-cylinder path method offer significant improvement in latency over the buffer node implementation, even though the latter also offers good improvement over the buffer-less 4-ary network. A slight better performance in throughput is also shown in the inter-cylinder path method. In summary, we conclude that the inter-cylinder path enhancement provides a more attractive solution over the buffer based solution for various network operation conditions, especially promising for low redundant and high load conditions.

Keywords- data vortex network; packet switched network; optical; network; buffering.

I. INTRODUCTION

Switching fabric networks are important subsystems in high capacity communication networks and computing systems. A typical space switch uses rich connectivity to handle dynamic traffic coming from a large number of input/output (I/O) ports while maintaining a high data throughput and small latencies. In high end multi-processor computing applications, the number of I/O ports or processors can reach thousands with each running at data

rates of tens of Gbit/s. At the same time low latency (tens or hundreds of μ s) must be maintained through such networks. Multistage self-routing network architectures often provide better system scalability, where each of the distributed routing nodes incorporates relatively simple routing logics. Such arrangement leads to cost-effective implementation and shorter delay due to simple processing at each stage. In order to provide higher data throughput, such networks can be implemented using optical fibre and optical switching technology.

Many recent researches have focused on developing optical switching fabric networks and network testbeds. In particular, this paper is a continuation of research presented in reference [1]. While it is relatively easy to achieve higher transmission bandwidth with Wavelength Division Multiplexing (WDM) within a single fibre, the routing logics and the handlings of traffic contention are hard to manage directly within the optical domain [2][3]. In particular, Data Vortex packet switched network architecture is developed for the ease of photonics implementation, and such networks are highly scalable to support a large number of I/O ports where each runs at high data rate and the network maintains a small routing latency [4]-[6]. The combination of its high spatial connectivity and an electronic traffic control mechanism among the routing nodes lead to bufferless operation and a much simpler routing logic within the nodes. Even though it uses deflection based routing, the spatial connectivity avoids large deflection penalty and reduces overall probability of deflection; therefore, it is advantageous compared with other commonly used interconnection architectures.

Previous researches on Data Vortex networks have focused on two main areas. One of the aspects has to do with physical implementation of the system. A small scale network testbed with 36 nodes and 12x12 I/O ports at Columbia University has been used to study various physical layer limitations. In particular as the number of node hops increases, optical signal to noise ratio (OSNR) and signal degradation were examined with various physical parameters. It has been shown that optical packets using an 8 wavelength payload at 10Gbit/s per channel can transverse 58 hops before a bit error rate (BER) of 10^{-9} is reached [7]. Therefore, the physical layer performance has shown promising scalability. Additional efforts are on switching device integration to support the size scalability. Current

testbed and system designs are based on semiconductor optical amplifier (SOA) switches because of their broad gain bandwidth and fast switching speed at nanoseconds, which is compatible with packet switching. Even though previous researches have not yet shown the same level as Data Vortex's potential sizes, several experimental works have demonstrated that a modular design can be used to build up a much larger matrix of SOA switches with required drivers and controls[8][9]. Integration related issues should be addressed for future study at much larger sizes and relevant cost scaling factor should also be explored in details. More recent researches on alternative switching devices based on silicon photonic technology can also provide potential solutions if these devices offer fast switching speeds while maintain low loss nature during the routing [10].

The second aspect focuses on enhancement in routing performance through network architecture designs. Although earlier researches have shown that with sufficient network redundancy, Data Vortex network scales to support a large number of I/O ports while achieving high throughput and low latency performance, at extremely high load conditions, and less redundant network conditions, the throughput tends to be limited by traffic backpressure in the deflection based routing. Therefore, network design researches may solve these issues with modified and enhanced functionality introduced in Data Vortex architecture. Simulation studies are typically conducted to examine the network performance under various traffic and operation conditions with different network sizes. There have been several approaches suggested to enhance the routing performance of the Data Vortex networks, especially for less ideal operating conditions [11]-[14]. In general, these performance enhancement methods require additional routing paths or routing resource, thus detailed cost and performance analysis must be carried out in comparison to the original network for a fair argument. There has been no comparison between different enhancement methods under the same operating condition, so this paper emphasizes such comparative study of two specific methods to contribute to the insights of the issues. The two methods, using node buffering and using extra inter-cylinder paths respectively, are of particular interests because they share the same cost with reasonable hardware increase in comparison to the original network. Among proposed, they are also relatively easy to implement thus more practical. The performance will be compared to each other as well as to the original Data Vortex networks. While random traffic is used for benchmark study, we also extend performance comparison under bursty traffics [15], which have not been previously studied within the enhanced networks. Simulation parameters are selected to focus on worse operation conditions such as low redundancy, high traffic load or bursty condition. In addition, recently a k -ary Data Vortex architecture based on multiple header bit processing at each stage has been proposed, which is shown to effectively reduce the latency when incorporated with buffer

implementation. This is mainly due to smaller number of cylinders thus the forwarding delay is kept small in comparison to the overall delay [6]. Therefore, we also extend the proposed comparison study between two approaches in a 4-ary Data Vortex network, and examine if the results for the original binary Data Vortex follow a similar trend in 4-ary networks.

The paper is organized as follows: in Section II, the original Data Vortex network architecture is explained in details. In Section III, two previously proposed enhancement methods, the nodal buffering method as well as inter-cylinder path method are illustrated and compared in details. The routing performance comparison is provided in Section IV for various network conditions, and the comparison is extended to bursty traffic conditions as well as to 4-ary Data Vortex networks. Finally the conclusion is given in Section V.

II. DATA VORTEX ARCHITECTURE

The Data Vortex architecture arranges its routing nodes in three dimensional multiple stage configuration as shown in Fig. 1. The size of the switching fabric is characterized by the height, H and angle, A of the cylinder. The number of cylinders is $C = \log_2 H + 1$ due to binary decoding routing process. The last cylinder is optional, but typically included to provide additional optical buffering for the output ports where electrical buffering is situated. Fig.1 shows routing path organization along each of the $C=5$ cylinders of the Data Vortex network with $A=4$, $H=16$. While the cylindrical levels ($c=0$ at the outermost cylinder to $c = \log_2 H$ at the innermost cylinder) provide the multiple levels in the routing stages, the angular dimension with repeated connection patterns provides multiple open paths to the destination therefore results in a much smaller latency penalty as deflection occurs. Inter-cylinder paths are not shown for a better view, and they are simply parallel links that maintain the height position of the packets when they propagate from outer to inner cylinders. These are used for forwarding purpose only between the different levels.

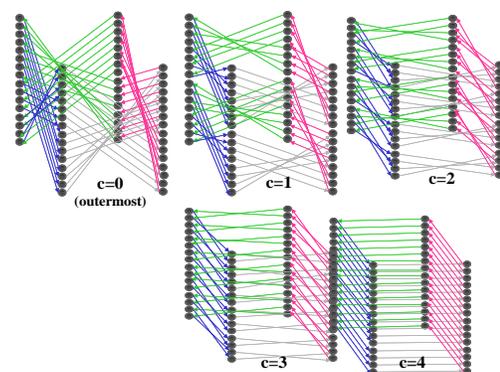


Figure 1. Data Vortex network with Angle=4, Height=16 and Cylinder=5 and its layout of routing node at different cylinders

A network in operation can connect I/O ports to active injection angle A_{in} with $1 \leq A_{in} \leq A$, and the ratio $\frac{A_{in}}{A}$ controls the network redundancy. For example, previous researches have shown that $\frac{A_{in}}{A} \cong \frac{1}{5}$ provides highly redundant condition, and it allows for each I/O port to reach above 95% injection rate even at full traffic load. This however requires an expensive implementation with $5 \times A_{in} \times C \times H$ number of routing nodes. Therefore, optimum choice of A_{in} should balance between the number of I/O ports and the desired routing performance.

Data Vortex networks operate in synchronous slotted fashion. Optical packets travel from the outermost cylinder to the innermost cylinder where the correct target height of the packet is located. To achieve simple self-routing process, each packet's destination height is encoded in binary. In the physical layer implementation, each of these binary bits is modulated onto a distinct wavelength, so that simple passive wavelength filtering can be used to extract and decode the single header bit h_i at the i^{th} cylinder level. This is shown within the node structure in Fig. 2. Only a small amount of optical power is tapped and converted from optical to electronics (O/E) for header decoding purpose. Majority of the packet and power stays in optical domain as it travels through the network. Each node accepts either *West* (*W*) input (from the same cylinder) or *North* (*N*) input (from the outer cylinder or from the injection port). Only a single input can be present at the same time through traffic arbitration. The packet is routed either to *East* (*E*) (to the same cylinder) or to *South* (*S*) (to the inner cylinder) by turning on the proper SOA switch (SW). Each SOA provides power amplification to balance the power loss at the node due to tap and 3-dB power splitter between *E* and *S* paths, and its broad spectrum and fast nanosecond switching speed are appropriate for packet switching operation. The payload data is modulated using WDM technique as well, so that a typical packet of hundreds of nanoseconds could provide enough information per packet.

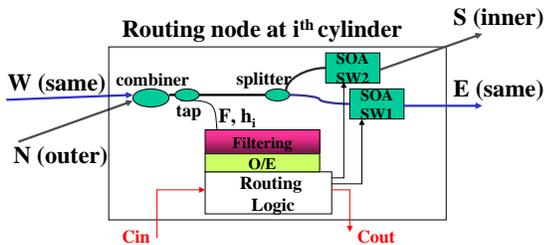


Figure 2. Routing node implementation

Data Vortex network combines a traffic control mechanism with deflection routing. Control signals stay in the electronic domain for simple implementation. As seen in the routing node in Fig. 2, a control signal C_{in} dictates whether *South* path to the inner cylinder is “open” or “blocking”. Each routing node also generates a proper C_{out}

to inform its outer cylinder neighbour node. The distributed control signal allows for the neighbouring nodes to coordinate properly and satisfy the single packet processing condition for each node. This can be illustrated in Fig. 3 in a triangle of routing nodes who shared the control signal path. Every time a packet is to stay at its current cylinder or to the *East* path, it creates a “blocking” control C_{out} for its outer cylinder contender. For example, if node A sends a packet to node B, it generates a “blocking” control for node C as shown in Fig.3. In the case the outer traffic receives a “blocking” control, the packet that is intended for *South* path will be deflected by staying on its current outer cylinder and wait for the next open path in two hops. In this example, packet of node C stays on cylinder $c-1$ until the next inter-cylinder path or corresponding control is open. The single packet routing arrangement eliminates optical buffers within the routing nodes as the network serves as virtual buffers as the packet travels on the cylinders.

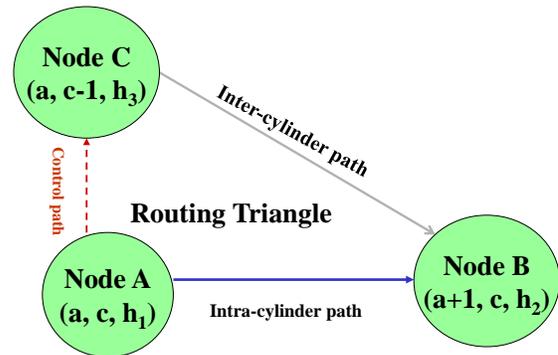


Figure 3. Control Signal in Routing Triangle

As mentioned, the last cylinder is typically added for optical buffering purpose so packets loop around the last cylinder at the same height position. Note that inter-cylinder paths and intra-cylinder paths are slightly different in length to allow for the establishment of the control signal and timing requirement. The inner cylinder nodes always make the routing decision slightly earlier than their outer neighbour to inform the traffic condition, so by making the inter-cylinder travel slightly shorter, packets arrive at the same node at the same time frame regardless of their origins. Detailed traffic control and routing performance have been reported in earlier studies [4]-[6]. Overall, Data Vortex networks maintain reasonable routing performance even as the networks scale up to thousands of I/O ports. In addition, many physical layer limitations have been studied and addressed in previous studies.

III. MODIFIED DATA VORTEX IMPLEMENTATION

As Data Vortex networks run at high load conditions or less redundant configurations, i.e., more input angles are attached to the I/O ports for incoming traffic, the traffic

backpressure could build up between the cylinders, so it takes longer to go through the network and the overall throughput also drops significantly. Due to the physical degradation of the optical signal through each node, reduction in latency is highly desired as well as maintaining the high data throughput. There have been several approaches suggested to enhance the routing performance of the Data Vortex networks with additional hardware. The detailed analysis of cost and performance comparison to the original network has been reported in earlier studies [11]-[13]. This paper emphasizes performance comparison of two methods using buffering and extra inter-cylinder paths respectively. Because the hardware increase in both methods is reasonably low and the costs are close to each other, a comparison of the two implementations under the same operation conditions is of great interests. In addition to previously reported random traffic performance, we have also extended the performance comparison for bursty traffic conditions. Section A provides an overview of the buffering method presented in [11], and section B provides an overview of the extra inter-cylinder path method presented in [12].

A. Buffering

The original Data Vortex network is attractive for its bufferless operation. However, for enhanced performance, separate buffers can be added within the routing nodes with slightly more complicated routing logic. This allows for less deflection when the packets wait in the buffer of the present node instead of circulating around the cylinders.

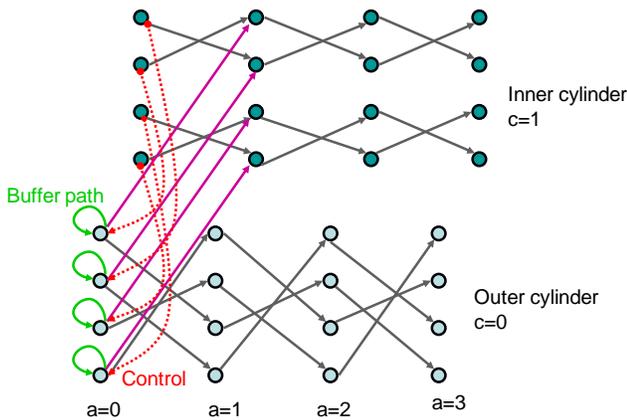


Figure 4. Data Vortex network with buffers within node shown at a=0.

Fig.4 shows the network implementation where nodes are arranged in the exact same fashion, except buffer paths are added within each node, as shown in an example for nodes at angle a=0. These buffer paths are simply delay lines with proper latency for routing purpose. The details of modified routing node are shown in Fig. 5. An additional switch (SW3) provides the third routing path to the buffer unit. Both the combiner and splitter will handle three potential inputs, so the splitting loss is slightly higher. The

single packet routing principle is maintained so that only three SWs are required. In order to inform the presence of the traffic within the buffer path to maintain the single packet routing principle, the buffer unit must have at least two slot delays to allow for correct set up in timing of the control signal.

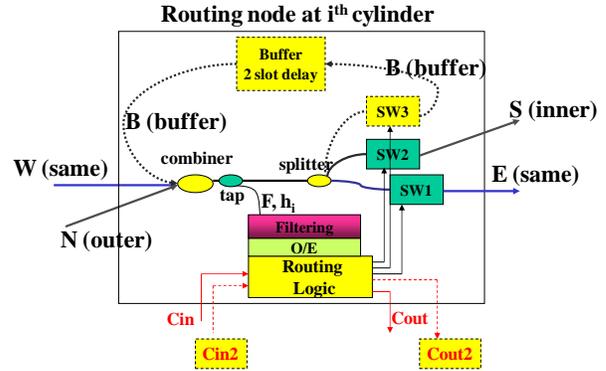


Figure 5. Routing node with buffer implementation: a 2-slot delay for buffer path is necessary to setup the control signal on time and additional controls C_{out2} are used to inform the state of buffer

Reference [11] also proposed a buffering scheme with a single slot delay, which is based on two simultaneous packets routing principle. While the routing performances are greatly improved, the required hardware is also significantly more because each node requires 6 SWs instead of 3 SWs. We are interested in a simpler and more cost-effective solution, so this study focuses on the buffer method shown in Fig.5 only that maintains a single packet routing principle through a two hop delay buffer. This implementation requires the network to have roughly 50% more hardware in number of switches and in routing paths compared to that in the original network. The modification of routing logic is minimal. If a packet is not able to reach S output, it will travel to the buffer unit and enter to the same node in two time slots. If the buffer packet is being processed, neither W nor N would accept inputs to maintain the single packet. As a result, priority is given to the packet within the buffer, and if there is no buffer traffic, then the same cylinder traffic gets the priority over the outer cylinder traffic as that in the original network. The additional control signal has to inform both the same cylinder neighbour and the outer cylinder neighbour to avoid contention.

B. Inter cylinder paths

In addition to buffering, there have been proposals for additional routing paths between the cylinders for enhanced routing performance [12][13]. The routing paths between the cylinders are critical resource and determine how fast traffic moves through the cylinders. Competition for these routing resource results in deflection thus builds up traffic backpressure. In this paper, we focus on the extra inter-cylinder path implementation as reported in [12], and a separate study has shown very similar enhancement results

for implementations in [12] and [13] under various traffic and network conditions. In the scheme shown in [12], we allow the packet to be routed to a secondary inter-cylinder path S_2 output if there is no other traffic (from regular *West* and *North* path) entering that same node. The addition the inter-cylinder path greatly improves the routing resource between cylinder levels. An additional injection path is also provided at each of the injection ports so that packets are less likely to be blocked by the traffic that is already circulating around the outermost cylinder. The setup of extra links and controls is shown in Fig. 6, and a detailed node implementation is shown in Fig. 7. The single packet routing rule is maintained for simplicity and an additional SOA switch (SOA-SW₃) is used to provide the third routing path as shown in the routing node. In this case, the additional control is necessary to inform the same cylinder traffic so that the traffic that goes to the regular S_1 output obtains the higher priority over the traffic that requires the S_2 output path. The secondary inter-cylinder path is of the same length as the original inter-cylinder path; therefore, it does not penalize packets that take the extra path in their delay. The implementation is merely trying to use the routing resource as much as possible while offer fairness to packets through the cylinders.

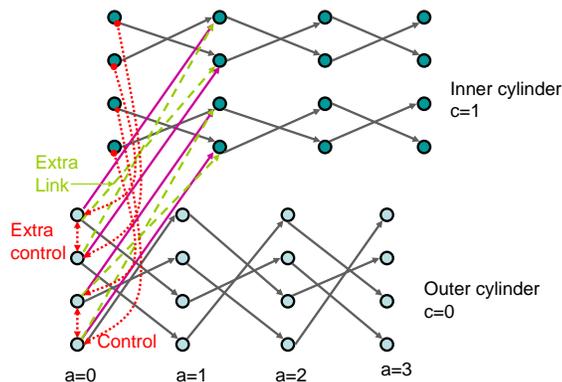


Figure 6. Extra inter cylinder path in Data Vortex network with required extra control

The height choice for the secondary inter-cylinder path must be such that the binary bits for all the previous cylinders maintain the same as those in the height of primary inter-cylinder path. As an example implementation, for a routing node at position of (a, c, h) , its S_2 path connect to a node $(a+1, c+1, h')$, where h' can simply invert the $(c+1)^{\text{th}}$ bit of h where both height in binary format. Therefore, the first c header bits are locked the same to maintain the routing progress from the current node to either S path or S_2 path. The inter-cylinder path implementation requires about 50% more hardware in the number of switches and number of routing paths; therefore, it has comparable cost to the buffering implementation shown in section A.

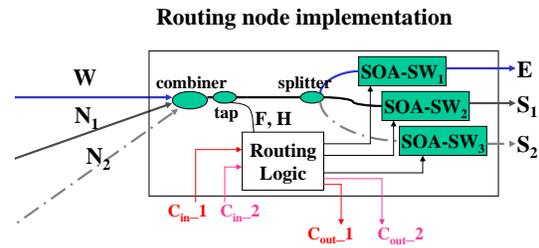


Figure 7. Modified routing node

IV. PERFORMANCE EVALUATION

In order to compare the effect of node buffering and inter-cylinder path for routing, a simulation in C/C++ is written to study the routing performance such as latency and data throughput. The compared networks are of the same size and same load conditions. The performance metric include average latency, latency distribution and network throughput. The average latency is measured for all the arrival packets for a long period of simulation time after the initial injection transient period. The network throughput is measured as the successful injection rate at the input port as previously reported. Latency distribution statistics are collected for arrival packets to see the range of the latency in packet switched operation. Once the packet reaches the correct target height, it exits the network immediately, therefore no angular resolution is considered in this simulation study. The performance evaluation extends beyond random traffic condition, and includes bursty traffic conditions as well as within a modified k -ary network implementation. These results are presented in section A, B and C respectively. The simulation runs sufficiently long for at least 5,000 clock cycles and the statistics are collected after steady state has been reached after the initial 500 clock cycles. All the results are presented with confidence level above 99% in comparison to a much longer simulation period or across various random seeds that are used to generate the traffic patterns. In all cases, the traffic load varies from 0.1 up to 1.0. Input angles A_{in} are typically chosen to be 3 or 5 to reflect medium to low redundant conditions in a network of $A=5$. Most simulations are carried out at a reasonably large size with $H=256$, and even higher sizes up to $H=1024$ are discussed for scalability study.

A. Performance comparison for random traffic

First random traffic pattern is studied to provide baseline performance. Random traffic indicates that each I/O port is independent, and they have a fixed probability of injecting packets, which depends on a set traffic load. Each packet slot also independently chooses its destination and its destination is uniformly distributed across all heights. Two enhancement methods are incorporated in a network of $A=5$, $C=9$ and $H=256$ as an example. Because both methods are for performance enhancement purpose when the Data Vortex network is heavily loaded or under less redundant operation, we choose the active injection angle to be $A_{in}=3$

and $A_{in}=5$ for the study. Keep in mind, for the buffer implementation, each buffer stay requires a two packet slots delay even though the number of node hop is one. The latency performance is measured in terms of packet slots to represent the physical delay.

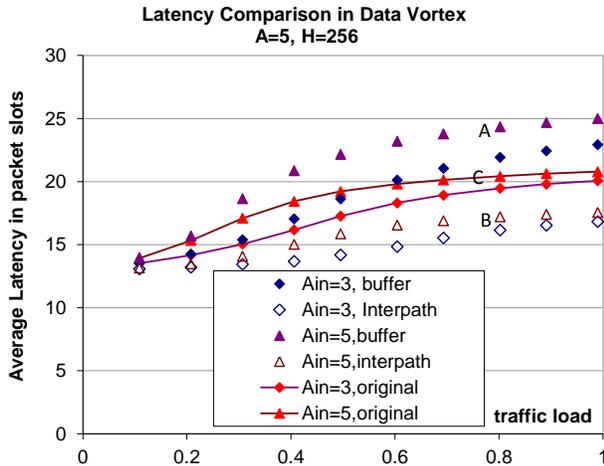


Figure 8. Latency comparison under various traffic load and redundant conditions

First, we examine the latency performance as shown in Fig. 8. For comparison purpose, the original network performances are shown as the solid lines. From these results, we can see that the inter-cylinder paths provide a smaller latency in general compared to that with an additional buffer within the routing node. In fact, the latency is worse for the case of node buffering compared to the original network especially at higher load conditions and less redundant network conditions. This is mainly because of the two hop delay requirement on the buffer path for timing requirement, which does not provide efficient reduction of latency even though the deflection events are reduced by keeping the packet at the open path to inner cylinder. The traffic backpressure remains significant because as the buffer packet re-enters the node for routing, there is no acceptance of additional traffic from neighbouring nodes. On the other hand, the inter-cylinder paths provide a better shared configuration of the redundant resource because when such resource is available, the additional routing paths always push more traffic through towards the inner cylinders. As a result, the traffic backpressure has been more effectively reduced. At the full load, the difference in latency in two methods is as large as 6 packet slots, which is 26.7% improvement if normalized.

The latency distribution is another important measure of the delay performance. In particular, we compare the latency distribution for $A=5$, $H=256$ with $A_{in}=5$ and at load of 0.8 for two implementation methods, i.e., network A and B shown in Fig. 8. The original network of the same condition or network C in Fig.8 is also shown as a reference. The latency distribution comparison is shown in Fig.9. A much

narrower distribution is achieved in the inter-cylinder path approach, which dramatically reduces the average latency as previously shown in Fig.8.

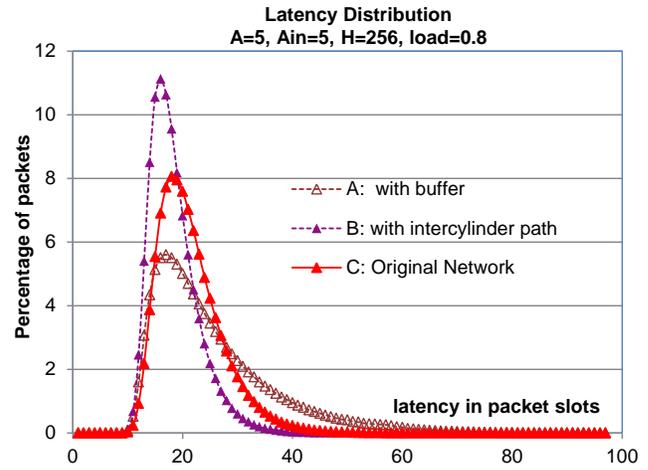


Figure 9. Latency distribution comparison for A, B and C in Figure 8.

The throughput performance comparison is shown in Fig. 10. A similar performance edge in inter-cylinder path implementation over buffer based implementation is reflected. In this rather busy network conditions, the buffer implementation has little improvement compared with the original networks, while the inter-cylinder path approach provides much more visible improvement. Both redundant conditions show very similar trend in comparison.

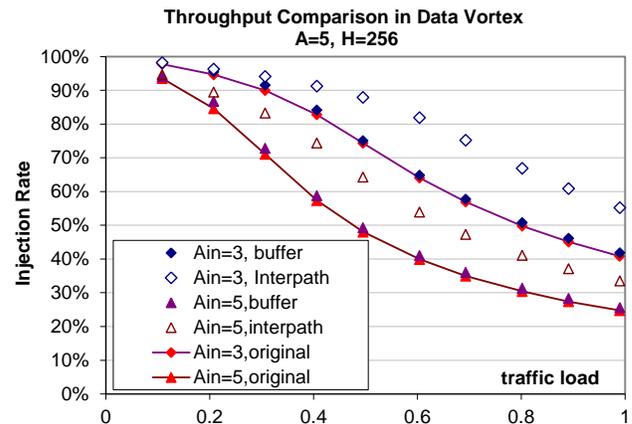


Figure 10. Throughput comparison under various load and redundant conditions

In reference [11], more detailed cost performance study is provided on this buffer implementation in comparison to the original network. Similar conclusion is provided that the overall the improvement in throughput and latency in this buffer scheme is rather limited and this implementation is only attractive for much lighter traffic conditions or more redundant networks. In our comparison for more heavily

loaded networks, the results have proved that the buffered implementation could even degrade the overall network performance once the system reaches saturation in load. On the other hand, the inter-path approach maintains the performance enhancement in both throughput and latency, and it provides a much more attractive solution for the same amount of hardware cost. Such performance enhancement clearly scales to very demanding network conditions as shown.

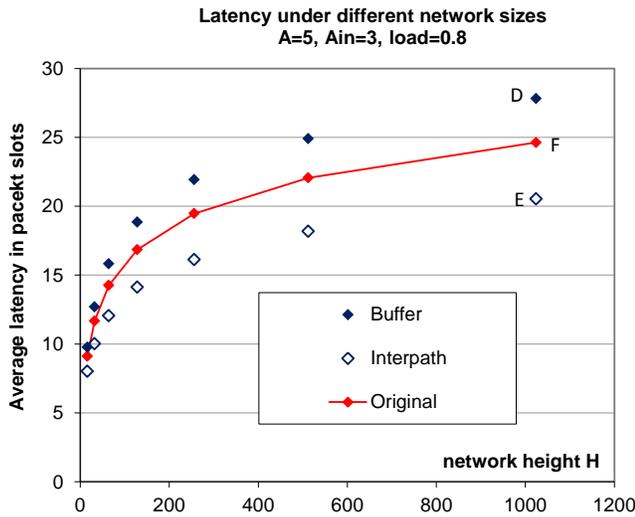


Figure 11. Latency performance comparison at different network sizes

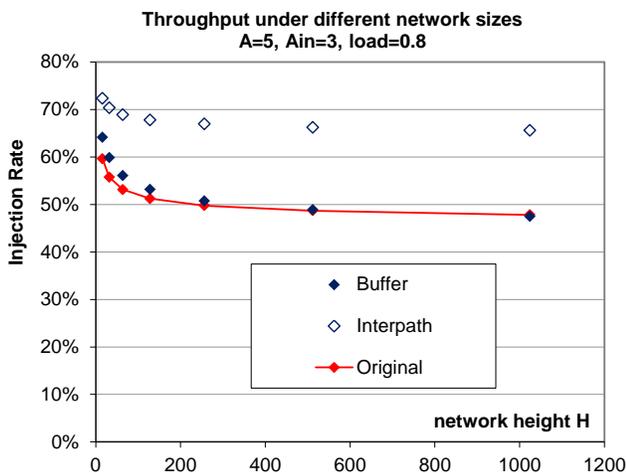


Figure 12. Throughput performance comparison at different network sizes

In order to study the scalability of such performance comparison, networks of different heights are also compared in the study. In Fig. 11 and Fig. 12, networks with both enhanced methods are compared with the original Data Vortex network with $A=5$ and $A_{in}=3$. All cases shown are with a medium to high traffic load of 0.8. It is shown that for all network sizes, the inter-path cylinder approach provides better performance over the buffer implementation, and there is especially significant difference for larger

networks. In the case of $H=1024$, the latency difference between two methods is as large as 7 packet slots, which is 26.1% improvement if normalized. The throughput difference is as high as 18%, which is an improvement of 27.7% when normalized.

Finally, the latency distribution comparison for the two implementations for network height of $H=1024$, i.e., D and E shown in Fig.11 are also compared, and the original network F of the same condition is shown as a reference. As seen in Fig.13, the inter-cylinder path method provides much narrower latency distribution, and thus results in a much smaller average latency. As packets stay within the network less time on average, overall higher traffic throughput are achieved at the same time.

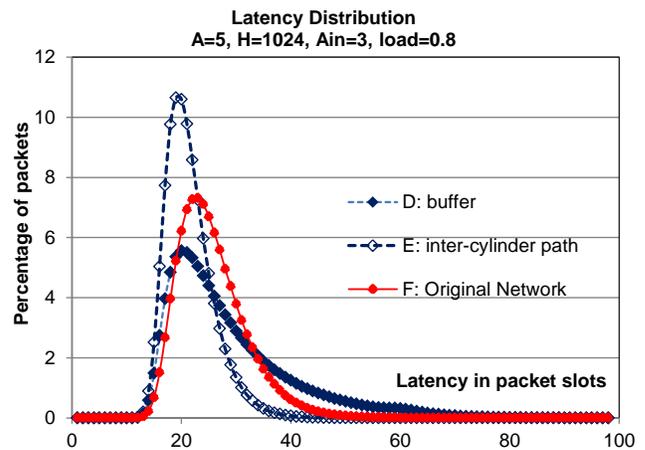


Figure 13. Latency distribution for D, E and F in Fig.11

B- Performance comparison for Bursty traffic

To show the performance comparison for even worse or more realistic traffic conditions, we extended the comparison to bursty traffic conditions. The effect of bursty traffic in the original Data Vortex has been studied thoroughly in reference [14][15], but the two enhanced networks with buffer and with inter-cylinder path have only been studied with random traffic. Here these enhancement networks will be subject to similar burstiness in traffic, and the results of modified architecture under the bursty traffic will be compared to each other, but also compared to random traffic performance.

As reported in [15], each ON period T_{on} and OFF period T_{off} are modeled by $T_{on} = \left\lfloor \frac{1}{U^{1/\alpha_{on}}} \right\rfloor$ and $T_{off} = \left\lfloor \frac{1}{U^{1/\alpha_{off}}} \right\rfloor$ respectively so that they follow rounded Pareto distributions. Here U is a random variable uniformly distributed over $[0, 1]$, and $\lfloor \cdot \rfloor$ indicates the floor function. Parameters $(\alpha_{on}, \alpha_{off})$ specify the length of the consecutive injection slots and length of consecutive idle time slots, where consecutively injected packets are also of the same destination and treated as a burst. Each input port is modeled independently and traffic loads are averaged over different

input ports during the total simulation time. Table I indicates the burstiness parameter ($\alpha_{on}, \alpha_{off}$) and corresponding traffic load conditions used in the simulation study. In comparison to a random traffic of the same level of load, each burst goes to the same destination instead of individual slot; therefore such traffic pattern also causes hot spot in routing if the network is not properly designed.

Table I: Bursty parameter and actual load

Bursty Parameter		Actual load
α_{on}	α_{off}	
1.05	8.0	0.856
1.05	2.5	0.815
1.5	5.0	0.712
1.5	2.5	0.655
5.0	5.0	0.5
5.0	1.5	0.29

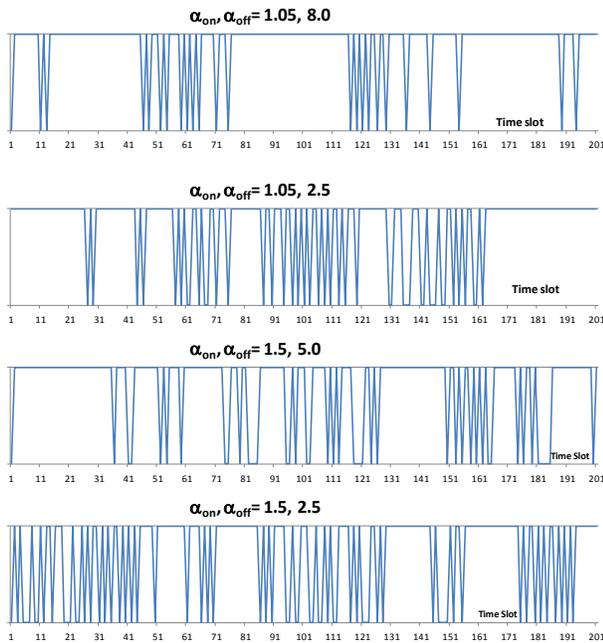


Figure 14. Bursty parameter and traffic patterns

Typical traffic patterns with the specified bursty parameters are shown in Figure 14 for comparison purpose. To really see the worst network condition, we present the comparison for the least redundant network condition with $A_{in}=5$.

Figure 15 and 16 show the latency and throughput performance respectively. As seen, the performance gain for inter-cylinder path implementation over buffer node

implementation is even more obvious with bursty traffic conditions. In particular, the latency in buffer node networks shows a much worse uptrend (purple solid triangle) as the load increases for bursty traffic. The inter-cylinder path network on the other hand shows a very similar performance in latency between random and bursty traffic even at much bursty or higher load conditions. They almost follow the same range with much smaller sensitivity to the increases in load or burstiness. The throughput performance gain shows slight edge in inter-cylinder path method, but the performance difference is less obvious than the gain in latency performance.

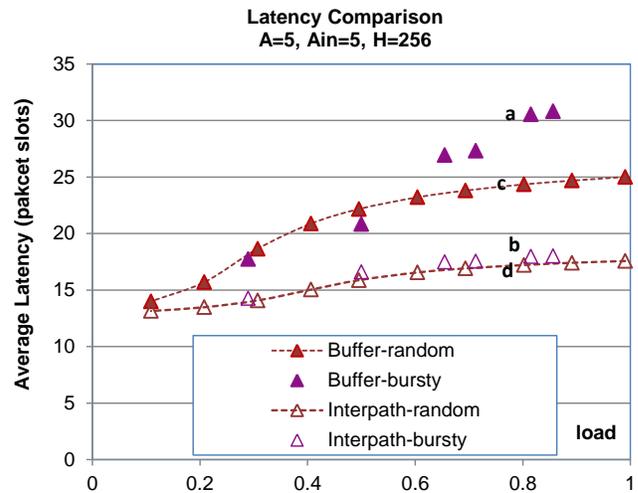


Figure 15. Latency performance comparison

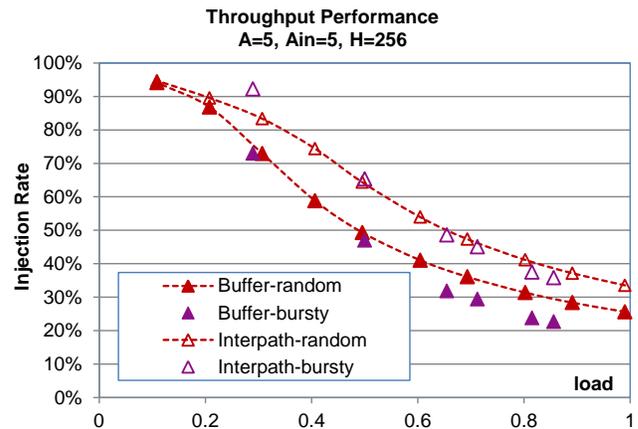


Figure 16. Throughput performance comparison

To further understand the latency performance, we also compare the latency distribution under various traffic conditions. In particular at load of 0.8 shown in Fig. 15, case *a* and *b* for bursty traffic and *c* and *d* for random traffic are compared and their latency distribution performance are represented in Fig. 17 and Fig. 18 respectively. As with the average delay, the distribution curve shows much narrower

range of packet latency with the inter-cylinder path implementation. On the other hand, with buffer implementation, the latency distribution shows much slower tail, and this is especially obvious in the case of bursty traffic conditions, which partially explained the much larger difference between case *a* and *b*, and this difference is more than difference between random traffic case *c* and *d* at a same level of traffic load.

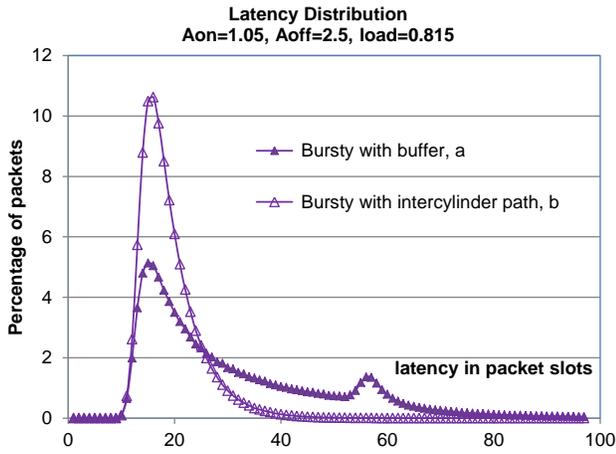


Figure 17. Latency distribution performance comparison

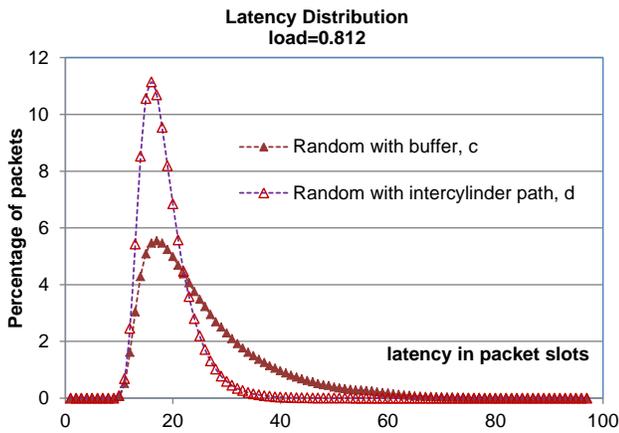


Figure 18. Latency distribution performance comparison

We also compared the performance difference for bursty traffic for different redundant conditions. As seen in Fig. 19 and 20, a similar trend is observed from a medium redundant network with $A_{in}=3$ in comparison to $A_{in}=5$ shown earlier. While the benefit is shown slightly less, it emphasizes the same conclusion that the inter-cylinder path implementation is more advantageous over buffer node implementation especially when the network is subject to worse traffic conditions or for load higher than 0.5.

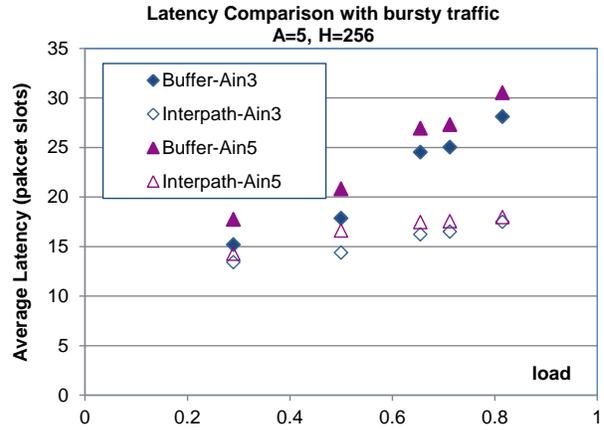


Figure 19. Latency performance comparison

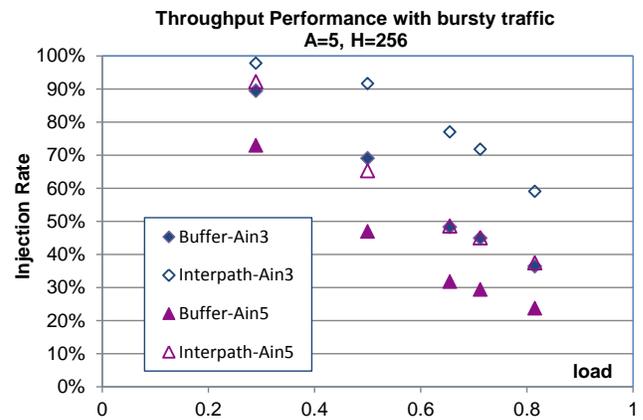


Figure 20. Throughput performance comparison

C- Performance comparison extended to 4-ary Data Vortex network

An alternative arrangement of *k*-ary Dava Vortex was proposed in a recent study, which is based on multiple header bit routing at each stage [16]. In particular, a 4-ary network ($k=4$) is shown to improve the latency performance due to the much smaller number of network cylinders and reduced forwarding latency. This is because number of cylinders is $C = \log_4 H + 1$ instead where each stage decodes two header bits ($\log_2 k = 2$) in a 4-ary network. When incorporated with buffer implementation, such arrangement shows particular advantages because of smaller deflection penalty in comparison to bufferless 4-ary network. Therefore, it is interesting to extend our comparison study between buffer implementation and inter-cylinder path implementation in the 4-ary Data Vortex networks. Whether there is a same level of difference in two methods in their enhancement in *k*-ary network should be an interesting extension to the comparison results in the original binary Data Vortex.

As an example, a 4-ary Data Vortex network is shown below in Fig. 21 which only requires $C = \log_4 H + 1 = 3$ cylinders for a network height of $H=16$. The routing node is modified as shown in Fig.22 so that the routing logic is based on two header bits and a similar traffic control mechanism is implemented to maintain the single packet processing principle. The routing path patterns of each cylinder can be constructed as shown [16].

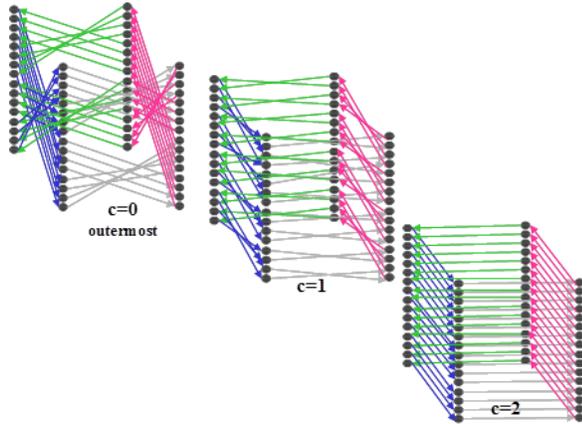


Figure 21. Routing patterns at each of the three cylinders in a 4-ary decoding Data Vortex network. $A=4, H=16, C = \log_4 H + 1 = 3$

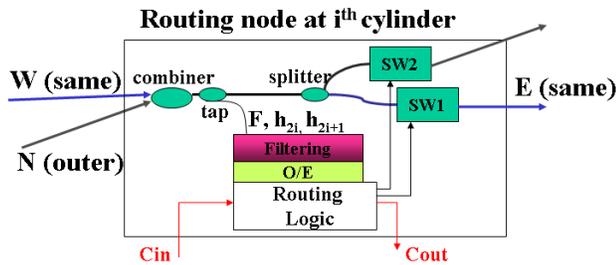


Figure 22. Routing node of 4-ary network that requires two header bits detection

The network comparison is carried out for a same network height of $H=256$ as that in the binary network study. $A=4$ is chosen for the symmetry of the routing path patterns on the cylinder. Two redundant conditions with $A_{in}=2$ and $A_{in}=4$ are compared for the study to focus on medium to low redundant network conditions. We also include the original 4-ary network without enhancement for reference, so the focus is on performance enhancement and comparison between two methods. Only random traffic is considered for this comparison.

The performance comparison in latency and throughput are shown in Fig. 23 and Fig. 24 respectively. A similar trend is observed in latency comparison, and under such redundant conditions, there is quite significant benefit of

inter-cylinder path implementation over the buffer node implementation. For example, at full load condition, with least redundancy $A_{in}=4$, the difference in two methods in latency is as high as 6 packet slots, which is 28% if normalized. When compared to buffer-less 4-ary network, the gain in inter-cylinder path also reaches 19.4%. As seen, the effect of node buffering becomes limited, and it does not provide enhancement as in more relaxed traffic conditions [16]. The significant improvement in inter-cylinder path shows its effectiveness in routing. From the throughput performance, the difference is less significant, but still the inter-cylinder path provides slightly more improvement in comparison to the original network. Both buffer and inter-cylinder path offers better throughput than the buffer-less 4-ary network, so traffic backpressure are reduced with both methods.

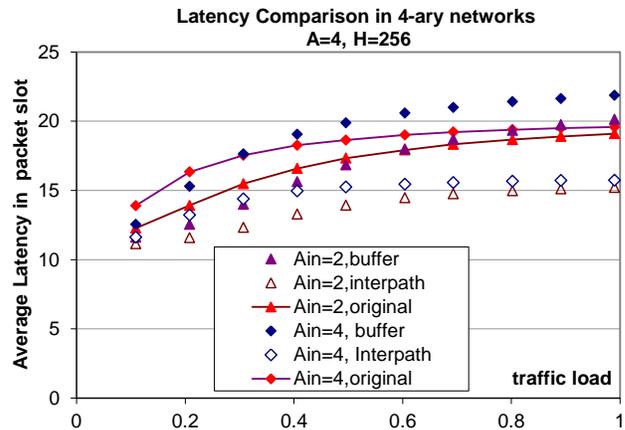


Figure 23. Latency performance comparison

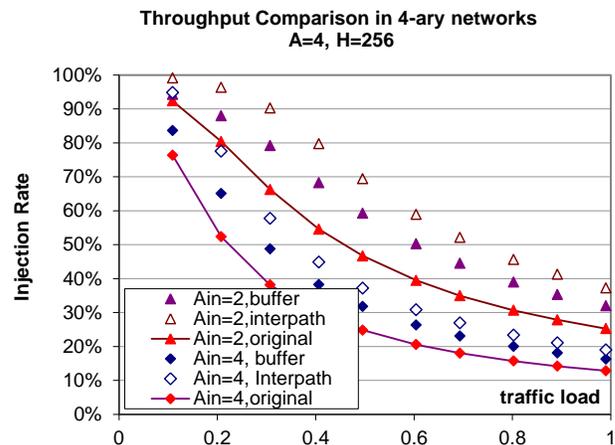


Figure 24. Throughput performance comparison

In summary, under medium to low redundant conditions, the 4-ary Data Vortex networks performance follows a very similar trend as that in the binary networks. Overall, the inter-cylinder path implementation provides much more

significant improvement than the buffer node implementation. It is especially beneficial shown in the latency performance due to its single slot nature of the extra inter-cylinder path while the buffer is based on two slots delay. The throughput performance also shows slight edge for the inter-path cylinder method. The 4-ary routing node implementation provides an overall reduction of the forwarding delay in comparison to binary network, but does not handle high traffic or less redundant conditions as well as binary which results lower throughput in general. The two enhancement methods provide greater benefits. Since the complexity and cost of implementation of two methods are the same, the inter-cylinder path offers a much more attractive solution because of its superior performance shown in all traffic conditions and network configurations.

V. CONCLUSIONS AND FUTURE WORKS

This study focuses on two different modification schemes for Data Vortex networks improvement. With similar hardware cost and complexity, the extra inter-cylinder paths provide a better configuration of the shared redundant routing resource. Such arrangement effectively reduces the traffic backpressure present in the original network at high load network conditions, and it provides much better performance in latency and data throughput than the modified network with buffering implementation. The extended study with bursty traffic further confirms the conclusion. The comparison in a 4-ary Data Vortex network shows a similar trend, and the inter-cylinder path method offers obvious benefit over the buffer method, particularly in latency performance. Future developments in switching device integration are important and relevant for this investigation, and allow us to further quantify the benefits of different enhancement schemes. For future development in novel enhancement methods, researchers should consider not only the hardware cost but also the routing performance in both delay and throughput, especially for less ideal network operation conditions. Only a thorough study provides a fair and effective evaluation of the proposed solution.

REFERENCES

- [1] Qimin Yang, "A Comparison Study on Data Vortex Packet Switched Networks with Redundant Buffers and with Inter-cylinder Paths", The third international conference on Emerging Network Intelligence (Emerging 2011), Lisbon, Portugal, November 20-25, 2011.
- [2] Keren Bergman, Optical Fiber Telecommunications, B: Systems and Networks (Editor Ivan P. Kaminow, Tingye Li, Alan E. Willner), Chapter 19, "Optical interconnection networks in advanced computing systems", Academic Press.
- [3] Ronald Luijten, Cyriel Minkenberg, Roe Hemenway, Michael Sauer, and Richard Grzybowski, "Viable opto-electronic HPC interconnect fabric", Proceedings of the 2005 ACM/IEEE SuperComputing, Seattle, pp. 18-18, November 2005.
- [4] Roberto Gaudino, Guido A. Gavilanes Castilo, Fabio Neri, and Jorge M. Finochietto, "Can Simple Optical Switching Fabrics Scale to Terabit per Second Switch Capacities?", Journal of Optical Communication Networks, Vol.1, No.3, pp. B56-B68, August 2009.
- [5] Odile Liboiron-Ladouceur, Assaf Shcham, Benjamin A. Small, Benjamin G.Lee, Howard Wang, Caroline P. Lai, Aleksandr Biberman, and Keren Bergman, "The Data Vortex Optical Packet Switched Interconnection Network", Journal of Lightwave Technology, Vol. 26, No. 13, pp. 1777-1789, July 2008.
- [6] Cory Hawkins, Benjamin A. Small, D.Scott Wills, and Keren Bergman, "The Data Vortex, an All Optical Path Multicomputer Interconnection Network", IEEE Transactions on Parallel and Distributed Systems, Vol. 18, Issue 3, pp. 409-420, March 2007.
- [7] Odile Liboiron-Ladouceur, Benjamin A. Small, and Keren Bergman, "Physical Layer Scalability of WDM Optical Packet Interconnection Networks", Journal of Lightwave Technology, Vol. 24, No. 1, pp. 262-270, January 2006.
- [8] A.Wonfor, H.Wang, R.V.Penty, and I. H. White, "Large Port Count High-Speed Optical Switch Fabric for Use within Datacenters", Journal of Optical Communication Networks, Vol. 3, No. 8, pp. A32-39, August 2011.
- [9] E.T. Aw, T. Lin, A. Wonfor, M. Glick, K.A. Williams, R.V. Penty, and I.H. White, "Layered Control to Enable Large Scale SOA Switch Fabric", 32nd European Conference and Exhibition on Optical Communication (ECOC 2006). Th.1.2.5, Cannes, France, September 24-26, 2006.
- [10] Xiaoliang Zhu, Qi Li, Johnnie Chan, Atiyah Ahsan, Hugo L. R. Lira, Michal Lipson, and Keren Bergman, "4 x 44 Gb/s Packet-Level Switching in a Second-Order Microring Switch", IEEE Photonics Technology Letters, Vol. 24, No. 17, pp. 1555-1557, September 2012.
- [11] Assaf Shcham and Keren Bergman, "On contention resolution in the data vortex optical interconnection networks", Journal of Optical Networking, Vol.6, pp. 777-788, June 2007.
- [12] Qimin Yang, "Enhanced control and routing paths in data vortex interconnection networks", Journal of Optical Networking, Vol. 6, No.12, pp. 1314-1322, December 2007.
- [13] Neha Sharma, D. Chadha, and Vinod Chandra, "The augmented data vortex switch fabric: An all-optical packet switched interconnection network with enhanced fault tolerance", Optical Switching and Networking, Elsevier, Vol. 4, pp. 92-105, June 2007.
- [14] Lianyong Dong, Qiang Dou, Quanyou Feng, and Wenhua Dou, "A Comparison Study of the Data Vortex Topologies with Different Parameter under Asymmetric I/O Mode", International Conference on Computer Science and Information Technology, pp.453-457, August 2008.
- [15] Qimin Yang, Keren Bergman, "Performances of the Data Vortex Switch Architecture Under Nonuniform and Bursty Traffic", IEEE Journal of Lightwave Technology, Vol. 20, No.8, pp.1242-1247, August 2002.
- [16] Qimin Yang, "Performance Evaluation of k-ary Data Vortex Networks with Bufferless and Buffered Routing Nodes", Asia Photonics and Communication Conference (ACP) 2009, pp. 1-2, Shanghai, China, November 2009.