Multi-Hop Scheduling Algorithms in Switches with Reconfiguration Latency

Multi-hop scheduling algorithms in switches with reconfiguration latency

Valentina Alaria*, Andrea Bianco†, Paolo Giaccone†, Emilio Leonardi†, Fabio Neri†
* CISCO System, USA Email: valaria@cisco.com
† Dipartimento di Elettronica, Politecnico di Torino, Italy, Email: {lastname}@tlc.polito.it

***Abstract*— Optical switching fabrics (OSF) are receiving increasing attention in the design of high speed packet switches, due to their excellent properties in terms of available bandwidth and reduced power consumption. However, most optical devices suffer a reconfiguration latency each time input/output connections are modified; unfortunately, this latency may not be negligible with respect to the packet transmission time, and can adversely affect performance, especially delay and throughput.**

The multi-hop approach, i.e., sending packets to the final destination port exploiting transmission to intermediate ports, was shown to be a promising way to control the tradeoff between delay and throughput. In this paper, we examine the multi-hop approach when using a logical interconnections based on multidimensional regular topologies. We discuss not only the scheduling problem for these topologies, but also the design of routing and queueing schemes. Performance are analyzed by simulation.

I. Introduction

[1] Hybrid optical/electronic switching architectures are considered today the most promising approach to design routers able to reach aggregate bandwidths up to 100 Tbps [1]. This is due to the intrinsic limitation of electronics at high speed, and to the photonic technology which is not mature to made all-optical packet switching a practical alternative in designing routers.

In hybrid opto/electronic switches, the switching fabric is fully optical and is typically located in a different rack with respect to the switch line-cards. Packets arrive at the router through optical links, and, after optical/electronic conversion, they are processed and buffered in the line-card; after an electronic/optical conversion, packets are sent over optical fibers to the optical switching fabric.

Regardless of the specific technology used to build optical switching fabrics (OSF), such as MEMS [2], bubble switches [3], broadcast and select networks with tunable devices [4], etc. a common feature is that whenever the OSF configuration (input/output ports connections) is changed, a *reconfiguration latency* is required before communication takes place due to reconfiguration constraints. We assume in this paper that all the ports of the switch are blocked during the reconfiguration phase. The reconfiguration latency is usually not negligible with respect to the packet transmission times (which are in the order of few μs at very high line rates); thus, it can adversely affect switch performance.

[1]This work was supported by the EU FP6 Network of Excellence e-Photon/ONe (through WP4)

As a consequence, the scheduling algorithm, whose task is to select the switching configuration of the optical device, should try to minimize the number of reconfigurations required to efficiently transfer a given traffic pattern. However, on the one hand to obtain high throughput, the scheduling should keep for long time the same switching configuration, so as to reduce the negative effect of inactivity periods due to the reconfiguration overhead; on the other hand, low delays imply to change quickly the switching configuration, so as to allow the full connectivity between all ports to be obtained in a short time interval.

To the best of our knowledge, few works have been proposed that consider the additional constraints due to reconfiguration latency when defining the scheduling problem (see [5], [6], [7], [8]). All these works assume that, when input i is connected to output j, only packets stored at input port i and destined to output port j can be transferred through the switching fabric, i.e. all the packets cross the switching fabric only once. In other words, when N packets are present at one input and destined to different N outputs, at least N switching fabric reconfigurations are required to allow the full connectivity between all inputs and outputs to be obtained, and to transfer N packets in sequence. In the worst case, the minimum access delay experienced at the head of the queue in an empty switch is $NT+\delta$, where T is the reconfiguration latency [9] and δ is the packet transmission time; this delay can be unacceptable for large switches. Thus, if we do not take a different approach, this simple observation may compromise the hopes towards the use of optical devices in routers in the future.

To overcome this problem, we exploit a *multi-hop* approach, which was proposed in [10], and later examined in [9]. The main idea of multi-hop scheduling is to configure the switching matrix once in a while, on a time scale significantly larger than packet transmission time, and to re-circulate packets among ports, i.e. a packet at input port i may reach its destination port j via successive transmissions through one (or more) intermediate ports. In the same scenario previously considered, the worst-case access delay for a multi-hop approach can be a much smaller value ($N\delta$) than the one obtained with the traditional single-hop approach; this delay can be acceptable for practical implementations.

Analytical modeling to discuss the properties of multi-hop approaches and the design of switches exploiting multi-hop scheduling was presented in [9]. In this paper we focus on

performance results obtained by simulation when considering a specific logical topology; moreover, we discuss fairness issues and how ports should be mapped to logical topology nodes depending on the switch traffic matrix.

II. The multi-hop approach

We assume that the switch has a synchronous behavior: fixed size packets are switched on a time slotted basis. The single all-optical switching fabric behaves as a buffer-less crossbar, i.e. at each time no more than one packet can be sent from an input port and can be received at an output port. A feasible switching configuration is equivalent to a *matching* in a bipartite graph, in which left-most nodes represent the input ports and right-most nodes the output ports; an edge connects left node i to right node j if input port i is connected to output port j.

We consider a switch with N ports, each running at the same line rate; all the packets arriving at the same input port and directed to the same output port belong to the same flow. Input queues are used to solve contentions among packets contending for the same output.

A centralized scheduling algorithm is in charge to select the sequence of appropriate switching configurations (matchings). Since, at each reconfiguration, a penalty in terms of latency has to be paid, to achieve high throughput the same matching must be held for a duration which is at least comparable with the reconfiguration latency.

To transfer all the packets according to a classical single-hop approach, full connectivity among switching ports is necessary (i.e., each input port has to be connected to every output port); as a consequence, the scheduling algorithm must cycle among at least N switching configurations. By doing so, however, the access delay can increase to unbearable values. On the contrary, according to the multi-hop approach, only a partial connectivity may be sufficient to guarantee the transfer of any packet through the switching fabric. Through a reduced set of switching configurations, input port i is directly connected by the scheduler only to a subset of other ports to which it can directly transmit packets. Packets directed to port j which is not connected to the port i, reach the destination in a multi-hop fashion, i.e. through some intermediate ports.

More formally, the multi-hop approach can be modeled in the following way. A connected *virtual interconnection topology* is overlaid to the set of switch ports; each node of the topology corresponds to a switch port. Let σ be the correspondence between the topology nodes and the switch ports; let $\sigma(i)$ be the node associated to port i. Consider now two generic nodes $\sigma(i)$ and $\sigma(j)$. If $\sigma(i)$ and $\sigma(j)$ are adjacent (i.e., it exists an edge in the virtual topology between them), then port i will be directly connected to port j by a proper matching chosen by the scheduler; indeed, the scheduling process is induced by the adopted topology. If node $\sigma(i)$ and $\sigma(j)$ are not adjacent, port i and port j will not be directly connected; however packets will flow from port i to port j though a set of intermediate ports which correspond to a path of the topology connecting node $\sigma(i)$ and node $\sigma(j)$.

Note that more than one path may connect two generic nodes, but we assume that a deterministic *routing algorithm* chooses only one of the possible shortest paths, to prevent mis-sequenced delivery of packets belonging to the same flow.

Depending on the chosen virtual topology and routing scheme, the scheduler selects the matching to transfer the packets from input to output ports. Let η be the efficiency of the switching fabric, defined as the fraction of time in which the switching fabric is available for packet transfer; it results $\eta = P/(P+T)$, being P the average holding time of matchings and T the reconfiguration latency. We denote with the term "epoch" a time interval comprising a matching holding time followed by reconfiguration time. As already stated, to achieve an high efficiency from the switching fabric, the same matching must be held for a duration which is larger than the reconfiguration latency.

We assume stationary traffic and we consider only periodic scheduling, in which a precomputed, fixed periodic scheme [11], [12] is adopted with *constant* holding time P. Given the overlaid virtual topology, the scheduler computes a minimal set of matchings, called *covering matchings*, which covers all the edges of the topology. Let M_c be the resulting number of matchings for a particular topology; note that M_c is equal to the maximum between the in-degrees and out-degrees of all the nodes of the topology, thanks to the Birkhoff von Neumann theorem [13]. These M_c matchings are sequentially selected to configure the switching fabric according to the periodic scheme. We define as *frame* a time horizon of length $M_c(P+T)$ in which a complete scheduling cycle is performed. Note that a fixed periodic frame scheduling allows an easier implementation at high speed and can be designed to support efficiently uniform traffic.

Two types of internal bandwidth speedup S are allowed. In the case of *temporal speedup*, the switching fabric runs S times faster than the line rate and during each epoch up to SP packets are served at each port; note that the frame duration remains the same. In the case of *spatial speedup*, S switching fabrics run in parallel (*spatial speedup*), configured with different covering matchings; thanks to this, the frame duration is reduced by a factor S; in addition, when $S = M_c$ there are enough switching planes to cover all the topology without reconfiguration and the reconfiguration latency is null: $T = 0$. Finally, note that temporal and spatial speedup can be also combined together.

In summary, to design an efficient multi-hop scheduler the following issues should be considered:

- definition of the virtual interconnection *topology* and its mapping to the switch ports;
- definition of a suitable *routing* strategy of packets on the virtual topology;
- definition of the *frame scheduling* plan.

Of course all the three previous issues are not independent. The definition of the virtual interconnection topology has a direct impact both on the definition of the scheduling plan and on the definition of the packet routing strategy.

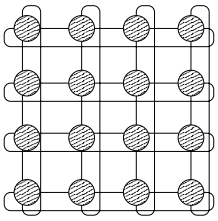

Fig. 1. Bi-dimensional Manhattan Street topology with 16 nodes

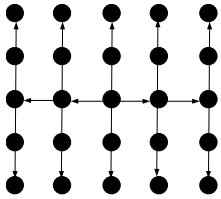

Fig. 2. Routing paths, according to PDR, for the central node of a 5×5 Manhattan network corresponding to a 25×25 switch; at most 2 directions are needed to reach any destination node.

In addition, to improve system performance, both the virtual interconnection topology and the routing strategy should be adapted to the traffic pattern. Indeed, the goal of the virtual interconnection topology and routing strategy design is to minimize the amount of packets sent in multi-hop fashion.

As a consequence, both the topology and the routing could be devised as adaptive to match actual traffic conditions to optimize performance, with the additional complexity of avoiding mis-sequenced packets. Since traffic is stationary, we consider only static schemes in which both topology mapping and routing are fixed.

III. Multi-hop for Manhattan topologies

Many interconnection topologies can be mapped onto the switching ports. Previous work [10] has considered ring topologies, whereas [9] has shown theoretically the advantages of multidimensional topologies. Among these, in this work we will consider only multidimensional, bidirectional regular square grid topologies, known in the literature with the name of Manhattan Street topologies. Fig. 1 shows an example of bi-dimensional topology. The main advantage of this topology is its simple definition, simple routing and good tradeoff between delay and throughput [9].

We consider as an example of the multi-hop approach the case of the Manhattan topology, overlaid to a 16×16 switch. Considering Fig. 1, each input/output port corresponds to a node of the Manhattan topology, according to the following bijective mapping: node (i, j), located in row i and column j, with $0 \leq i,j \leq 3$, corresponds to port $k = 4 \times i + j$, $0 \leq k \leq 15$.

Given that we rely on a regular topology with node degree 4, port $k = (i, j)$ can directly (i.e. in single-hop) reach four ports: $k_1 = (i, |j+1|_4)$, $k_2 = (i, |j-1|_4)$, $k_3 = (|i+1|_4, j)$, $k_4 = (|i-1|_4, j)$; [2] all the other destinations must be reached in a multi-hop fashion. The scheduling frame is partitioned in four fixed epochs: in the first scheduling epoch every node (i, j) is connected to $(i, |j+1|_4)$ for a time equal to P and we say that the direction followed in the topology is "down"; in the second scheduling epoch, every node (i, j) is connected to $(i, |j-1|_4)$, for a time equal to P (following "up" direction); in the third scheduling epoch every node (i, j) is connected to $(|i+1|_4, j)$, for a time equal to P (following "right" direction); in the forth scheduling epoch every node (i, j) is connected to $(|i-1|_4, j)$ for a time equal to P (following "left" direction). In this case the frame duration is $4P + 4T$; each scheduling epoch $P + T$ is associated to a specific direction and, hence, to a matching.

[2]We denote with $|\cdot|_n$ the modulo-n operator, i.e., the remainder of the division by n.

This example can be extended to multidimensional Manhattan topologies of generic dimension c, with degree $2c$ at each node; in this case, each side of the corresponding hypercube is $\sqrt[c]{N}$ nodes and the frame duration is $2c(P+T)$. Note that a bidirectional ring topology is obtained by setting $c = 1$.

Many routing algorithms on a Manhattan network can be devised. In our work we consider the following routing scheme, called "Privileged Directions Routing" (PDR), described for a bi-dimensional Manhattan network for simplicity, but that can be easily extended to multidimensional networks. Among all the possible shortest paths from a node (i, j) to a node (k, l), consider the path through node (i, l), following (possibly) first the row direction and then (possibly) the column direction. Fig. 2 shows the minimum distance routing paths followed by the central node of a 5×5 Manhattan topology to reach all other nodes. Note that the PDR scheme has the following properties: (i) unique routing path between any pairs of nodes, (ii) easy computation and (iii) the load across all the edges is balanced under uniform traffic.

Packets could be stored according to a classical Virtual Output Queue (VOQ) scheme: in the example of Fig. 2, the "right" direction allows the central node to reach 10 destinations nodes; hence, during the matching corresponding to the "right" direction the packets present in 10 VOQs are served. However, to allow fair and easier access to the switching fabric, we adopt a FIFO selection among all the packets served in the same input; this is equivalent to consider just $2c$ queues per input, one for each possible direction, instead of N VOQs at each input port, with evident benefits for the scalability of the queueing system for large switches.

A. Theoretical performance

In this section, we recall some performance bounds obtained in [9]. An upper bound λ_{up} to the maximum throughput under uniform traffic is given by the traffic load at which the link load equals the link capacity. [9] shows easily that:

$$\lambda_{up} = \frac{P}{P+T} \frac{1}{E[d]} \tag{1}$$

where $E[d]$ is the average overall nodal distance between two generic ports, according to the selected routing strategy.

Under PDR routing, the average distance for bidirectional Manhattan topologies can be estimated in the following way: for each dimension, two possible directions can be chosen, hence $\sqrt[c]{N}/4$ is the approximated[3] average distance traversed along the same direction; since c dimensions are allowed:

$$E[d] = c\sqrt[c]{N}/4 \tag{2}$$

The maximum throughput for multi-hop Manhattan (MH) topologies under uniform traffic, obtained by combining (1) and (2), is:

$$\lambda_{MH} = \frac{P}{P+T}\frac{4}{c\sqrt[c]{N}} \tag{3}$$

In the case of single-hop (SH), we consider a frame scheduling approach, adopting a sequence of N disjoint matchings given by the Birkhoff von Neumann decomposition [13] of the traffic matrix. Under uniform traffic, a frame is composed by N scheduling epochs; during the k-th scheduling epoch ($0 \leq k < N$), input port i will be connected to output port $|i+k|_N$ for a duration P. Hence, the maximum throughput for single-hop under uniform traffic is:

$$\lambda_{SH} = \frac{P}{P+T} \tag{4}$$

Note that a spatial speedup equal to $2c$ (independent from N) is sufficient for MH to obtain the maximum throughput without paying any reconfiguration penalty; whereas, for SH, the spatial speedup required is N. In the following sections, we will refer only to the effects of the temporal speedup.

For multidimensional Manhattan topologies, we can estimate easily an upper bound on the average *access delay*, i.e. the delay experienced by a generic packet entering an empty switch. This is a good approximation on the delay experienced at low loads. Observe that $2c$ (twice the degree of the topology) corresponds to the number of different matchings to provide full connectivity; hence, the frame lasts $2c(P+T)$. Furthermore, the shortest path between two generic nodes can be associated with an ordered sequence of h directions (one for each dimension, hence $1 \leq h \leq c$), corresponding to h different matchings. The average number of directions $E[h]$ taken by a packet is given by:

$$E[h] = \begin{cases} \dfrac{2(N-N^{1/2})}{N-1} & \text{for } c=2 \\ \dfrac{3(N-N^{2/3})}{N-1} & \text{for } c=3 \end{cases}$$

which can be shown by simple geometrical reasonings. On average, the time between two successive matchings corresponding to the same direction is $(2c-1)(P+T)$; hence, packets following a shortest path will experience, in the worst case, an access delay:

$$\bar{D}_{MH} = E[h](2c-1)(P+T) \tag{5}$$

[3]Precise evaluation of the average distance is possible, but this approximation gives an upper bound good enough for our purposes.

Parameter	Symbol	Value
reconfiguration latency	T	$0.12\ ms$
scheduling period	P	$1.2\ ms$
I/O link rate		10 Gbps
packet size		1500 bytes

TABLE I

PARAMETERS CONSIDERED FOR THE SWITCHING ARCHITECTURE UNDER STUDY

Algorithm	λ_{up}	Worst-case access delay (ms)
SH-N25	0.91	16.5
MH-2D-N25	0.36	6.6
SH-N27	0.91	17.8
MH-3D-N27	0.40	13.7

TABLE II

THEORETICAL PERFORMANCE UNDER UNIFORM TRAFFIC FOR SMALL SWITCHES

which is an upper bound holding for generic Manhattan topologies.

For single-hop, the average access delay $\bar{D}_{SH}$ can be bound simply by:

$$\bar{D}_{SH} = N(P+T)/2 \tag{6}$$

IV. SIMULATIVE PERFORMANCE STUDY

Table I shows the parameters considered in the switch under study. T is given by technological constraints related to MEMS reconfiguration latencies [8], and P is set to guarantee a switching efficiency $\eta \approx 90\%$, corresponding to 10% of throughput reduction in the single-hop case. With the packet set equal to the MTU of 10-Gigabit-Ethernet, the slot duration is $1.2\ \mu s$, corresponding to $T = 100$ timeslots and $P = 1000$ timeslots.

Let λ_{ij} be the average traffic load from input port i to output port j. We assume that $\lambda_{ii} = 0$, i.e., we do not send across the switching fabric a packet arrived at port i and directed to the same port i: in this case, we suppose, as in [1], that a dedicated data-path is present in the linecard to shortcut these packets from the input interface to the output interface of the same port. If now ρ is the offered traffic load for a generic input, we considered the following traffic scenarios, for $0 \leq i, j < N$:

- **Uniform**: $\lambda_{ij} = \dfrac{\rho}{N-1}$, for $i \neq j$.
- **Lin-diagonal**: $\lambda_{ij} = \dfrac{2\rho}{N(N-1)}|j-i|_N$.

The size of each VOQ is set equal to 400 packets.

A. Performance under uniform traffic

Fig. 3 shows the average delay with respect to the offered load ρ for single-hop (SH) and two scenarios:

- multi-hop Manhattan bi-dimensional (2D) topology, with $N = 25$;
- multi-hop Manhattan tri-dimensional (3D) topology, with $N = 27$.

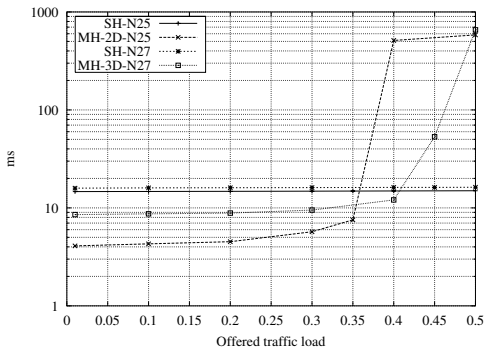

Fig. 3. Average delay under uniform traffic for bi/tri dimensional multi-hop topologies (2D for $N = 25$ and 3D for $N = 27$)

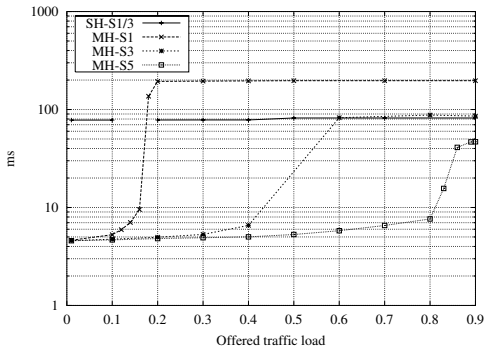

Fig. 4. Average delay under uniform traffic, for large switch $N = 121$, 2D multi-hop topologies, and variable speedup S

Both scenarios refer to small size switches. Table II shows the maximum throughput and the worst-case access delay estimated by the theoretical models of Section III-A. In Fig. 3, SH shows almost a flat delay with respect to ρ, since its performance is dominated by the frame scheduling; the average delay is tightly bounded by the worst-case access delay. Multi-hop schemes, both 2D and 3D, shows delay quite well estimated by the theoretical models, when $\rho < \lambda_{up}$. When the offered load becomes non admissible, the delays are dominated only by the queueing process inside the finite-size buffers.

This graph confirms our expectations: when increasing the dimension of the topology (from 2D to 3D), the maximum throughput increases but at the expenses of a larger access delay.

The throughput reduction due to multi-hop schemes can be compensated by speedup. Fig. 4 shows the average delay for a large switch ($N = 121$) for single-hop and 2D multi-hop schemes, in the case of variable speedup $S = 1,\ 3,\ 5$. Also in this scenarios, the average delay for low load and the maximum throughput are well bounded by the approximated models of Section III-A, reported in Table III.

It is important to observe that the average delay for single-hop is not affected by the speedup, because of the scheduling approach based on frames; of course, single-hop can benefit

Algorithm	λ_{up}	Worst-case access delay (ms)
SH-S1	0.91	80.0
SH-S3	1.00	80.0
SH-S5	1.00	80.0
MH-2D-S1	0.17	7.3
MH-2D-S3	0.50	7.3
MH-2D-S5	0.83	7.3

TABLE III

THEORETICAL PERFORMANCE UNDER UNIFORM TRAFFIC FOR LARGE SWITCHES ($N = 121$)

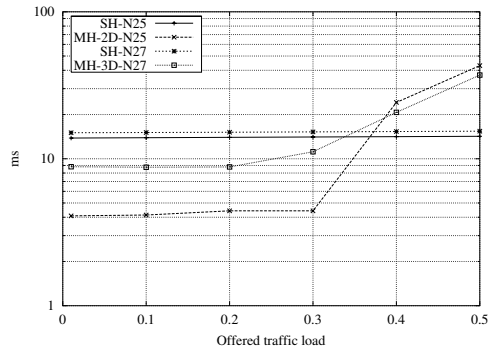

Fig. 5. Average delay under Lin-diagonal traffic, for 2D and 3D multi-hop topologies and small switches

of a very small speedup ($1/\eta = 1.1$) to achieve the maximum throughput.

On the contrary, for multi-hop throughput is affected by speedup: a speedup S is able to increase the maximum achievable throughput by a factor S. The delay for low load is independent from the speedup, since the reasoning to estimate the worst-case access delay is not affected by the speedup.

B. Performance under Lin-diagonal traffic

The conclusions drawn in the previous section do not depend on the specific traffic scenario considered, but hold also for non uniform traffic. Indeed, Fig. 5 shows the effects of 2D and 3D multi-hop schemes under Lin-diagonal traffic; higher

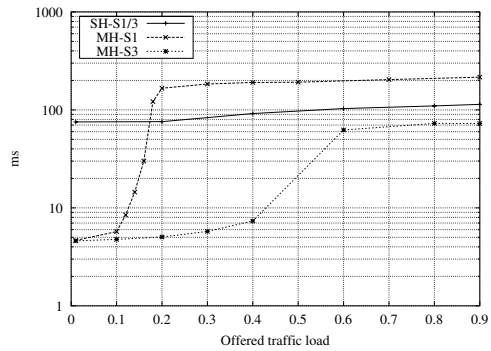

Fig. 6. Average delay under Lin-diagonal traffic, for 2D topology and variable speedup S, and switch size $N = 121$

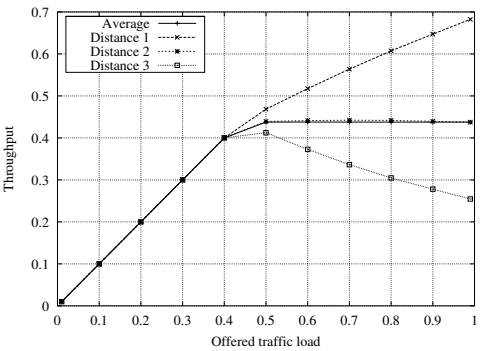

Fig. 7. Throughput under uniform traffic on a 3D multi-hop topology with $N = 27$

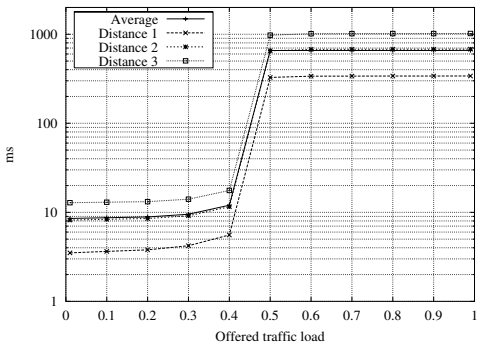

Fig. 8. Average delay under uniform traffic on a 3D multi-hop topology with $N = 27$

dimension of the topology means higher achievable throughput but at the expenses of a larger access delay (which is the same for uniform traffic).

Fig. 6 shows the speedup effect under Lin-diagonal traffic. The same qualitative behavior found for uniform traffic holds.

C. Unfairness in performance

When a packet experiences multi-hop transfer across the switching fabric, it contends many times for the access. Hence, we can expect better performance for packets experiencing less hops. This implies unfair performance between packets entering the same input port and directed to different output ports.

Fig.s 7 and 8 investigate the unfairness issue, showing, for $N = 27$, the throughput and the average delay achievable under uniform traffic for different classes of packets, each one associated to a particular distance (in terms of hops) from its input port to its output port. Table II reports that the 3D multi-hop scheme achieves 0.40 throughput and 9.9 ms access delay, averaging over all possible distances. This is in accordance with the curves labelled "Average" present in both figures.

Packets with lower distance experience lower average delay. At the same time, they experience higher throughput when the switch is overloaded; indeed, when considering many packet flows entering a work conserving queueing system, throughput may be unfair between flows only when the system capacity is overloaded.

V. Conclusions

In this paper we studied the multi-hop approach to schedule the packets across a switching fabric with very large reconfiguration latency. The main idea is to send a packet from an input port to an output port across the switching fabric through (possibly) many intermediate ports, in order to reduce the need of switching reconfiguration to provide full connectivity between input and output ports.

We have investigate the special case of multi-hop based on Manhattan topologies, and have shown the tradeoff between throughput, speedup and delays; this tradeoff can be quite well estimated by simple formulas, which have been confirmed by our simulation results.

Main finding of our investigation is that, especially for large switches, multi-hop approach can be convenient when delays are the main performance issues to consider.

References

[1] I. Keslassy, S.T. Chuang, K. Yu, D. Miller, M. Horowitz, O. Solgaard, N. McKeown, "Scaling internet routers using optics", *ACM SIGCOMM*, 2003, Aug. 2003, pp. 189-200.

[2] P.D. Dobbelaere, K. Falta, S. Gloeckner, "Advances in integrated 2D MEMS-based solutions for optical network applications", *IEEE Optical Communications*, May 2003, pp. 16-23

[3] S. Hengstler, J.J. Uebbing, P. McGuire, "Laser-activated optical bubble switch element" *2003 IEEE/LEOS International Conference on Optical MEMS*, Aug. 2003, pp. 117-118

[4] S.L. Danielsen, C. Joergensen, B. Mikkelsen, K.E.Stubkjaer, "Optical packet switched network layer without optical buffers", *IEEE Photonics Technology Letters*, vol. 10, n. 6, Jun. 1998, pp. 896-898

[5] K. Kar, D. Stiliadis, L.T. Lakshman, T.V., L. Tassiulas, "Scheduling algorithms for optical packet fabrics" *IEEE JSAC*, vol. 21, n. 7, Sept. 2003, pp.1143-1155

[6] I. Keslassy, M. Kodialam, L.T. Lakshman, D. Stiliadis, "On guaranteed smooth scheduling for input-queued switches", *INFOCOM 2003*, vol. 2, Apr. 2003, pp. 1384-1394

[7] B. Towles, W.J. Dally, "Guaranteed scheduling for switches with configuration overhead", *IEEE/ACM Trans. on Networking*, vol. 11, n. 5, pp. 835-847, Oct. 2003

[8] Xin Li; Hamdi, M.; "On scheduling optical packet switches with reconfiguration delay", *IEEE JSAC*, vol. 21, n. 7, Sept. 2003, pp. 1156-1164

[9] V. Alaria, A. Bianco, P. Giaccone, E. Leonardi, F. Neri, "Design of switches with reconfiguration latency", *IEEE ICC 2006*, Jun. 2006

[10] Andrea Bianco, Paolo Giaccone, Emilio Leonardi, Fabio Neri, Paola Rosa Brusin, "Multi-hop scheduling for optical switches with large reconfiguration overhead", HPSR 2004 (High Performance Switching and Routing), Phoenix, AZ, USA, April 2004

[11] T. Weller, B. Hajek, "Scheduling nonuniform traffic in a packet-switching system with small propagation delay", *IEEE/ACM Transactions on Networking*, 1997

[12] A.Bianco, M.Franceschinis, S.Ghisolfi, A.Hill, E.Leonardi, F.Neri, R.Webb, "Frame-based matching algorithms for input-queued switches", *High Performance Switching and Routing (HPSR 2002)*, Kobe, Japan, May 2002

[13] C.S. Chang, W.J. Chen, H.Yi Huang, "Birkhoff-von Neumann Input Buffered Crossbar Switches", *IEEE INFOCOM 2000*, Tel Aviv, Israel, Apr.2000, pp.1614-1623.

[14] Yeung, K.L.; Yum, T.-S.P.; "Node placement optimization in ShuffleNets", *IEEE/ACM Transactions on Networking*, vol. 6 , n. 3, June 1998, pp. 319-324