Scheduling Schemes for Delay Graphs with Applications to Optical Packet Networks

Isaac Keslassy* Murali Kodialam† T. V. Lakshman† Dimitrios Stiliadis †

∗ Computer Systems Laboratory
Stanford University
Stanford, CA 94305-9030
keslassy@stanford.edu

† Bell Laboratories
101 Crawfords Corner Road
Holmdel, NJ 07733
{muralik, lakshman, stiliadi}@bell-labs.com

***Abstract*— We revisit the problem of scheduling optical packet switched networks such as packet switches, WDM rings, and general mesh structures, when propagation delays are not negligible. We use a general model based on delay graphs, and show that several optical packet switched networks can be modeled using this abstraction. We devise mechanisms that take into account propagation delays and can achieve 100% throughput for admissible traffic patterns, when traffic demands are known. We then present necessary and sufficient conditions for providing 100% throughput in such graphs.**

I. Introduction

Optical packet/burst switching over Wavelength Division Multiplexing (WDM) has been extensively studied over the last few years and the technology has been applied to packet fabrics, ring architectures and mesh networks. The evolution of optical components with devices such as fast tunable lasers [10], [19] or large Arrayed Waveguide Grating Routers (AWGR) [2] make such architectures feasible.

For example, packet fabrics can be built with tunable transmitters, AWGRs and fixed receivers, or with fixed transmitters, optical star couplers and tunable receivers [10], [13]. Ring networks can be also built using similar techniques. In TTFR rings, transmitters are tunable and receivers are fixed; traffic is transmitted in terms of packets or bursts, and communication between two nodes is achieved by tuning the wavelength of the transmitter to the fixed wavelength associated with the receiver [18]. The second alternative (FTTR) assumes that transmitters are fixed and receivers are tunable and is equivalent to a "broadcast-and-select" architecture. Receivers can choose the transmitter that they want to listen to, and traffic is transmitted again in bursts. A third alternative allows both transmitters and receivers to be tunable providing further flexibility. In mesh networks, one can assume similar models where each receiver is associated with a wavelength and transmitters are tunable. A mesh of buffer-less optical switches interconnects the nodes. A node can transmit a packet to any other node by utilizing the appropriate wavelength.

If we abstract the operation of all the above architectures, we can identify the common requirements for correct operation. Data from two transmitters cannot arrive to one receiver at the same time and a transmitter cannot send data to two receivers at the same time. In addition data for the same receiver must not conflict at any point of the network. Therefore, some contention resolution mechanism is necessary to prevent conflicts.

In addition, the above technologies are often applied in network settings, with relatively long propagation delays; for example, a single packet switched network within a campus, or a metro-area ring, or a core optical switched network. In these cases, propagation delays are an important component of the scheduling problem. Our goal is to study the structural properties of the propagation delays and show necessary and sufficient conditions for achieving 100% throughput in such networks.

Although the scheduling problem has been formally studied before in most of the above settings, and with different parameters [13], [4], [12], [11], [17], [5], [6], the problem of variable propagation delays has not been addressed in a generalized form. In this paper, we revisit the problem of scheduling in such applications, by using a generalized model of delay graphs. Based on this model, we study large optical packet switched networks and show that in all of these cases we can achieve 100% throughput when propagation delays are decomposable. Further, we present a mechanism for achieving this structure in general optical nodes.

II. Problem Formulation

A. Notations and Assumptions

We first introduce some notations, describe the different problem settings, and show how these different settings can be modeled in a similar fashion. Consider the general model of Figure 1. A set of transmitters and receivers are interconnected through some buffer-less optical network. The network can be a ring, a star, or some other non-blocking structure (i.e. every permutation of arrivals on the receivers can be satisfied, as long as there are no output conflicts). u_i and v_j denote propagation delays to some intermediate network node k, and without loss of generality we assume that delays are expressed in terms of integer timeslots.

Mesh – In the most general setting, we can assume that the interconnection network is a *mesh* connecting N edge nodes numbered $\{1, 2, ..., N\}$. Each node has a *fast Tunable Transmitter and a Fixed Receiver (TTFR)*. The fast tunable transmitter is an optical device that can be made to transmit in different wavelengths. The fixed receiver is an optical device that always receives data using the same wavelength. Therefore the fixed receiver of a node j, where $1 \leq j \leq N$, will always receive data at an associated wavelength λ_j, with N distinct

∗ This work was done while the author was with Bell Labs.

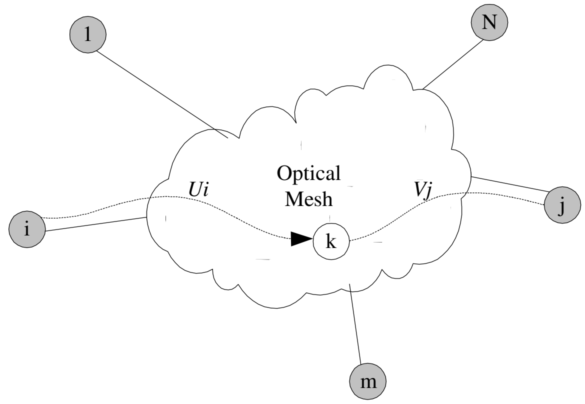

Fig. 1. General network model

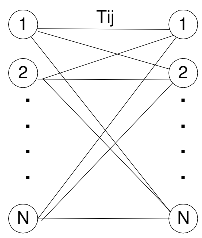

Fig. 2. Bipartite delay graph

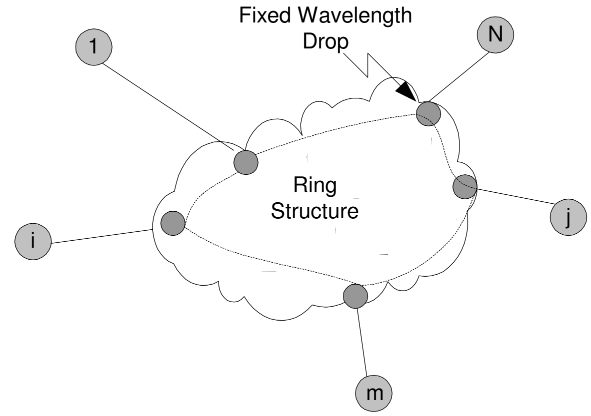

Fig. 3. WDM packet switched ring with tunable transmitters and fixed receivers

wavelengths associated to the N distinct nodes. In order for node i to transmit data to node j, it tunes its tunable transmitter to the wavelength λ_j, and then transmits data. Since node j is the only node listening to wavelength λ_j, it will be the only one to receive this packet. Therefore, a given wavelength is always associated with a specific receiving node, preventing contention for wavelengths between two different receiving nodes. In addition, the path taken by packets is assumed to be pre-determined by a routing algorithm, which ensures that the propagation time between any two nodes is the same for all packets. This assumption implies that if two packets do not collide at a shared intermediate node, they will also not collide at the destination node.

Bipartite delay graph – Let us model the mesh as an abstract $N \times N$ weighted bipartite graph. Input and output vertices are representing the N nodes, and edge weights are representing the propagation delays between the nodes. Thus, the edge from input vertex i to output vertex j has weight τ_{ij}. We will call such a graph a *bipartite delay graph* (Figure 2).

Separable graph – In the simpler case of packet switches based on AWGRs [10], [2] and broadcast systems based on star couplers, the propagation delays typically consist of two consecutive stages: from the source node to the hub, and from the hub to the destination node. It is possible to model these two stages as follows. First, each node i sends packets to a central hub, taking a propagation time u_i. Then, this hub switches the packet and sends it to its destination node j, taking an additional time v_j (possibly including some fixed processing time). Therefore, there exists two sequences u and v such that

$$\tau_{ij} = u_i + v_j \text{ for all } i, j. \tag{1}$$

Since we can separate the delays of this graph into two distinct components, we will call it *a bipartite delay graph with separable delays*, or simply a *separable graph* (Figure 1).

***F*-Separable graph –** Consider a uni-directional ring with N nodes and a total round-trip-time of F (Figure 3). For any nodes i and j, the union of the paths from node i to node 1 and from node 1 to node j is either equal to the path from node i to node j (if node 1 is on this path), or to the path from node i to node j augmented by the full round-trip of the ring. Therefore, the propagation time between node i and node j is always equal, modulo F, to the sum of the propagation time between node i and node 1 and the propagation time between node 1 and node j. Defining $u_i = \tau_{i1}$ and $v_j = \tau_{1j}$ for all nodes i, j, we get:

$$\tau_{ij} = u_i + v_j \bmod F \text{ for all } i, j. \tag{2}$$

We will call any graph that satisfies this equality *a bipartite delay graph with separable delays modulo F*, or simply an *F-separable graph*. For instance, any bi-directional ring of round-trip-time F is an F-separable graph as long as the routing policy involves no backtracking in the middle of a path. In addition, any separable graph is also F-separable. Note, that an F-separable graph is a generalization of the separable graph that allows the inclusion of ring architectures in our model abstraction.

B. Problem Definition

As is customary, we will assume for simplicity that time is slotted, and that all delays are multiples of a time-slot. Variable-length packets arriving to the mesh are segmented into fixed-length packets (or simply *packets*), and departing packets are reassembled into variable-length packets again. In addition, inside the mesh, each node can receive (respectively transmit) at most one packet per time-slot.

Rate matrix – We will assume that the network needs to periodically schedule a given rate matrix $R = [R_{ij}]_{1 \leq i,j \leq N}$. R_{ij} is the rate of packets arriving to node i (from outside the mesh) and destined to node j. This rate matrix could be explicitly given to the node (for instance in SONET), or could be periodically re-computed (for instance with data traffic), with a computation period significantly larger than the propagation and schedule times. In any case, we will assume that the rate matrix R is - or can be made - non-oversubscribed, i.e. doubly sub-stochastic, so that a node would not need to transmit (and receive) more than one packet per time-slot. Thus, for any two nodes i, j:

$$\begin{cases} \sum_{k=1}^{N} R_{ik} \leq 1, \text{ and} \\ \sum_{k=1}^{N} R_{kj} \leq 1. \end{cases} \tag{3}$$

Finally, we will assume that R is rational, so as to always be able to schedule it in a finite period. For instance, in SONET transport networks, traffic demands in the network can be based on an STS-1 granularity. The rate matrix R is then expressed as the ratio of an integer matrix by F, the number of STS-1 circuits. As an example, at OC192 speeds, R will be equal to the ratio of an integer matrix by $F = 192$, where all rows and columns of the integer matrix sum to at most F so that R is not over-subscribed.

Schedule – A schedule S of period F is a collection of F matrices of size $N \times N$ that are to be scheduled periodically. $S_{ij}(t)$ indicates the number of packets sent by node i to node j at time-slot t. Our objective is to find a schedule S of period F that satisfies the following four conditions, for all $1 \leq i, j \leq N$ and $0 \leq t \leq F-1$.

$$\begin{cases} (i) & \sum_{k=1}^{N} S_{ik}(t) \leq 1 \\ (ii) & \sum_{k=1}^{N} S_{kj}((t-\tau_{kj}) \bmod F) \leq 1 \\ (iii) & \frac{1}{F}\sum_{t=0}^{F-1} S_{ij}(t) \geq R_{ij} \\ (iv) & S_{ij}(t) \in \{0,1\} \end{cases} \tag{4}$$

(i) The first equation shows that each transmitter sends at most one packet per time-slot.

(ii) The second equation states that each receiver receives at most one packet per time-slot. Packets received by node j from node k at time-slot t were sent at time-slot $(t - \tau_{kj})$. By periodicity, the schedule at $(t - \tau_{kj})$ is the same as the schedule at $(t - \tau_{kj}) \bmod F$.

(iii) The third condition imposes that the demand for rate matrix R is satisfied.

(iv) The last equation indicates that at any time-slot, there is either one or no packet sent by a given transmitter at a given wavelength.

III. Time-Shifted Scheduling

A. Birkhoff-von Neumann Scheduling

The key complexity for finding an appropriate schedule stems from the inherent propagation delays. Indeed, consider the hypothetical case where *all delays are null* ($\tau_{ij} = 0$ for all $1 \leq i, j \leq N$). Our objective is to find a schedule S that satisfies Equation 4. Conditions (i),(ii) and (iv) of Equation 4 imply that at each time-slot t, $S(t)$ is a $0-1$ matrix with at most a single 1 per row and a single 1 per column. For instance, it could be a permutation matrix, i.e. a $0-1$ matrix with exactly one 1 per row and per column. Additionally, condition (iii) imposes that the sum of the schedules over any period should be greater than or equal to R. Given these constraints, it is known that it is possible to schedule the rate matrix R using a *Birkhoff-von Neumann (BvN) decomposition*. This decomposition is specified in the following theorem, shown in [5], [6] and based on [3], [16]. It can be computed using a polynomial time edge-coloring algorithm in a bipartite multigraph [9], [14], [1].

Theorem 1: Let R be an admissible rational rate matrix of common denominator F. Then there exists a minimal integer $K \leq \min(N^2 - 2N + 2, F)$, a set of positive integers $(\phi_k)_{1 \leq k \leq K}$, and a set of permutation matrices $(S_k)_{1 \leq k \leq K}$, such that:

$$R \leq \frac{1}{F}\sum_{k=1}^{K} \phi_k S_k, \text{ and } \sum_{k=1}^{K} \phi_k = F. \tag{5}$$

In other words, it is possible to schedule each matrix S_k for ϕ_k time-slots, such that the resulting set of F schedule matrices will satisfy Equation 4. We will call such a schedule *a BvN schedule*.

B. Time-Shifted Scheduling

Theorem 1 shows that a BvN schedule always exists when there are no propagation delays. However, if delays are introduced in a mesh, the scheduling problem becomes more complex. Can we generalize the BvN schedule to other graphs? When can we guarantee that we will be able to find such a schedule in a mesh? How can we find it? The following sections attempt to answer these questions.

Consider a separable graph defined as in Equation 1. A packet sent from any node i to node j will first arrive to some intermediate hub after a delay of u_i, and then to node j after an additional delay of v_j. Therefore, any packet sent from another input i' that could potentially collide with this packet when arriving at node j would also collide at the hub. As a consequence, making sure that packets do not collide at the hub is enough to make sure that they do not collide at the destination node. Therefore, by computing a BvN schedule at the hub, and then working backwards for each source node i in order to get the schedule u_i time-slots earlier, it is possible to make sure that there are no collisions. We call this the *Time-Shifted Scheduling (TSS)*.

TSS scheduling can be applied to F-separable graphs. More specifically, the following theorem proves that the set of graphs to which it is possible to generalize BvN schedules of period F includes all the F-separable graphs.

Theorem 2: Consider an F-separable graph, and a rate matrix R with BvN schedule S' of period F. Define the TSS schedule S as:

$$S_{ij}(t) = S'_{ij}((t + u_i) \bmod F), \tag{6}$$

where $1 \leq i, j \leq N$, $0 \leq t \leq F-1$, and u comes from Equation 2. Then the TSS algorithm can schedule R in this F-separable graph.

Proof: Let's show that the TSS schedule satisfies all the conditions of Equation 1 in order to prove the theorem. Conditions (i), (iii) and (iv) of Equation 1 result directly from the definition of the BvN schedule. Condition (ii) can be written as:

$$\sum_{k=1}^{N} S_{kj}((t - \tau_{kj}) \bmod F)$$

$$= \sum_{k=1}^{N} S'_{kj}(\{((t - [u_k + v_j]) \bmod F) + u_k\} \bmod F)$$
$$= \sum_{k=1}^{N} S'_{kj}((t - v_j) \bmod F)$$
$$= \sum_{k=1}^{N} S'_{kj}(t') \text{ where } t' = (t - v_j) \bmod F$$
$$\leq 1 \text{ (using condition } (ii) \text{ for the BvN schedule).}$$

■

The following corollary proves that the set of graphs to which it is possible to generalize all BvN schedules includes all the separable graphs. In other words, in separable graphs, it is possible to *guarantee a schedule to all rate matrices* R.

Corollary 3: Consider a separable graph and a rate matrix R. Then the TSS algorithm can schedule R in this separable graph.

Proof: Apply Theorem 2, and use the fact that any separable graph is F-separable. ■

IV. Theorems on F-Separable and Separable Graphs

A. F-Separable Graphs

We will now show that the *only* architectures in which it is possible to generalize the Birkhoff-von Neumann decomposition of frame period F are the bipartite graphs with separable delays modulo F. In other words, after we found in the last section that it is *sufficient* for the graph to be F-separable, and proved it using the TSS algorithm, we will now prove that it actually is a *necessary* condition.

Theorem 4: The following assertions are equivalent:

(i) It is possible to schedule any rate matrix R, where R has a (minimal) BvN decomposition of period F.

(ii) $\tau_{ij} = \tau_{i1} + \tau_{1j} - \tau_{11} \bmod F$ for all i, j.

(iii) The graph is F-separable.

Proof: Assuming (iii), (i) is possible using the TSS algorithm (Theorem 2).

Also, assuming (ii), (iii) is clear using $u_i = \tau_{i1} - \tau_{11}$, $v_j = \tau_{1j}$, and therefore $\tau_{ij} = u_i + v_j$.

Let's assume (i) and show by contradiction that necessarily (ii) is satisfied in order to prove the theorem. Otherwise, there would exist i, j and w_{ij} such that

$$\tau_{ij} = \tau_{i1} + \tau_{1j} - \tau_{11} + w_{ij} \bmod F,$$

with $w_{ij} \neq 0 \bmod F$. Consider the matrix R with $R_{11} = R_{ij} = 1/F$ and $R_{i1} = R_{1j} = (F-1)/F$. R looks like:

$$R = \frac{1}{F}\begin{pmatrix} 1 & 0 & 0 & F-1 \\ 0 & 0 & 0 & 0 \\ F-1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

First, node 1 and node i need to send exactly one packet at each time-slot, because the sum of the rates on rows 1 and i is 1, and they cannot send more than one packet per time-slot. Now, consider a periodic scheduling of period F (the proof similarly extends to any arbitrary scheduling by using iteratively the proof below). Since source node 1 sends packets to destination node 1 at rate $1/F$, it sends exactly one packet in each period. Similarly from node i to node j. For instance, assume source node 1 sends a packet to destination node 1 at the periodic times satisfying $t = 0 \bmod F$. Since the packet arrives at $\tau_{11} \bmod F$ to destination node 1, node i couldn't have sent any packet to node 1 at $\tau_{11} - \tau_{i1} \bmod F$. But node i has to send a packet at any time, and thus it sent it to node j. This packet will reach node j at $\tau_{11} - \tau_{i1} + \tau_{ij} \bmod F$, and therefore node 1 couldn't have sent a packet to node j at $\tau_{11} - \tau_{i1} + \tau_{ij} - \tau_{1j} \bmod F = w_{ij} \bmod F$. But node 1 has to send a packet at any time, and thus sent it to node i. Hence, summing up, if node 1 periodically sends a packet to node 1 at times $0 \bmod F$, it also needs to send a packet to node 1 at times $w_{ij} \bmod F$, with $w_{ij} \neq 0 \bmod F$. This is not possible because node 1 should only send at most one packet per period, hence contradiction. ■

B. Separable Graphs

Theorem 4 states the conditions needed to guarantee 100% throughput to all rate matrices with a BvN decomposition of period F. For example, in a SONET transport network at OC192 speeds with an STS-1 demand rate granularity, the rate matrix R is (or can be made) non-oversubscribed with a period $F = 192$. Then, in order to guarantee 100% throughput for any rate matrix, Theorem 4 states that the graph needs to be F-separable.

However, we are more generally interested in the conditions needed to guarantee 100% throughput for rate matrices with *any* frame period F. We can obtain the following key theorem for scheduling rate matrices in bipartite delay graphs:

Theorem 5: It is possible to schedule *any* rate matrix R if and only if the graph is separable.

Proof: Simply apply Theorem 4 for all F. Since for all F, $\tau_{ij} = \tau_{i1} + \tau_{1j} - \tau_{11} \bmod F$, then $\tau_{ij} = \tau_{i1} + \tau_{1j} - \tau_{11}$ (because the difference can be divided by any positive integer, and thus is null). Hence, the graph has separable delays. ■

This theorem is significant. It states that given all the assumptions defined at the start, the only way that a network operator can have a 100% throughput guarantee for any rate matrix R is to have a separable mesh network.

V. Generalization to Non-Separable Graphs

A. Optimization Problem

Consider a bipartite graph with non-separable delays. We will call this a *non-separable graph.* Theorem 5 proves that it is not possible to schedule all possible rate matrices R in this graph. Therefore, we would like to transform this graph such that any rate matrix R could be scheduled. In other words, we would like to make a non-separable graph separable. We will call this *a separable extension of the non-separable graph.*

In the remainder, we will assume that it is possible to add a specific delay δ_{ij} to each connection between source node i and destination j. For instance, the transmitter of each node i could be followed by the following devices. First, a $1 \times N$ AWG passive optical device that splits the incoming packets to different fibers according to their wavelength. Then, for each fiber associated to a specific node j and its wavelength λ_j, a delay line of delay δ_{ij} that adds a delay of δ_{ij} to each packet. Then, an

$N \times 1$ AWG that combines again the different wavelengths into a single WDM fiber. Since the delay lines are located in the source nodes, our assumptions on the mesh definition are still satisfied.

Using the additional delay lines δ_{ij}, our objective is to study how to make the non-separable delays separable, while minimizing the amount of delay lines used. In other words, our optimization function can be written mathematically as follows: $\min_{\delta} \Big(\sum_{i,j} \delta_{ij} \Big)$, such that:

$$\begin{cases} (i) & \hat{\tau}_{ij} = \tau_{ij} + \delta_{ij} & \forall i, j \\ (ii) & \hat{\tau}_{ij} = u_i + v_j & \forall i, j \\ (iii) & \delta_{ij} \geq 0 & \forall i, j \end{cases} \qquad (7)$$

The objective function shows that we want to minimize the sum of all delay lines for all the connections. The first condition expresses that the delays $\hat{\tau}_{ij}$ in the new system are equal to the delays in the old system, augmented by the new delay lines. The second condition states that the new system has separable delays. Finally, the last condition means that the added delay lines cannot be negative.

B. Optimal Separable Extension of Non-Separable Graphs

Let's solve the optimization function by rewriting it. We want to minimize:

$$\sum_{i,j} \delta_{ij} = \sum_{i,j} (\hat{\tau}_{ij} - \tau_{ij}) = \sum_{i,j} (u_i + v_j - \tau_{ij}).$$

Since the τ_{ij}'s are fixed, we are thus reduced to minimizing

$$\sum_{i,j} (u_i + v_j) = N \cdot (\sum_i u_i + \sum_j v_j).$$

The three conditions from the optimization problem can then be combined into a single one, yielding the following equivalent problem:

$$\min_{u,v} \Big[N \cdot \Big(\sum_i u_i + \sum_j v_j \Big) \Big], \text{ s.t.: } u_i + v_j \geq \tau_{ij} \quad \forall i, j \qquad (8)$$

In other words, we want to find the smallest separable delays that are still larger than each of the current delays. This optimization formula is surprisingly the same as another formula in another field. It is the dual of the maximum weight bipartite matching problem (Corollary 3.5.b. in [15]). As a consequence, the total minimum amount of delay lines needed for a separable extension of a graph with non-separable delays is:

$$N \cdot W_{MWBM} - \sum_{i,j} \tau_{ij},$$

where W_{MWBM} is the weight of a Maximum Weight Bipartite Match in the delay matrix τ.

Thus, this brings another way of looking at separable and non separable graphs. The separable graphs are the bipartite delay graphs for which all matches have the same maximum weight. On the contrary, some matches in non-separable graphs have strictly smaller weight than the maximum weight match. However, the smallest the weight difference, the less delay lines will be needed in order to have a separable extension of the non-separable graph, and therefore in order to be able to guarantee 100% throughput for any rate matrix R.

VI. CONCLUSION

We revisited the problem of scheduling in optical packet switched networks, and we showed that when propagation delays are taken into account, a variety of topologies and architectures can be modeled by a delay graph. Based on this model, we presented a simple algorithm that can achieve 100% throughput for all admissible traffic demands. Furthermore, we proved that a necessary and sufficient condition for achieving a guarantee of 100% throughput in these graphs is that the delays are separable, and showed how this can be achieved.

REFERENCES

[1] N. Alon, "A simple algorithm for edge-coloring bipartite multigraphs," *Information Processing Letters*, vol. 858, pp. 301-302, 2003.

[2] P. Bernasconi, C. Doerr, C. Dragone, M. Capuzzo, E. Laskowski and A. Paunescu, "Large N x N waveguide grating routers", *Journal of Lightwave Technology*, Vol. 18, No. 7, pp. 985-991, July 2000.

[3] G. D. Birkhoff, "Tres observaciones sobre el algebra lineal," *Universidad Nacional de Tucuman Revista*, Serie A, vol. 5, pp. 147-151, 1946.

[4] M.S. Borella, B. Mukherjee, "Efficient scheduling of non-uniform packet traffic in WDM/TDM local lightwave network with arbitrary transceiver tuning latencies," in *IEEE J. Select. A. Commun.*, Vol. 14, No. 5, pp. 923–934, 1996.

[5] C.S. Chang, W.J. Chen, and H.Y. Huang, "On service guarantees for input-buffered crossbar switches: a capacity decomposition approach by Birkhoff and Von Neumann," *IEEE IWQoS, London*, 1999.

[6] C.S. Chang, W.J. Chen, and H.Y. Huang, "Birkhoff-Von Neumann input-buffered crossbar switches," *Proceedings of IEEE INFOCOM '00*, Tel Aviv, Israel, pp. 1614-1623, 2000.

[7] C.S. Chang, D.S. Lee and Y.S. Jou, "Load balanced Birkhoff-von Neumann switches, part I: one-stage buffering," *IEEE HPSR '01*, Dallas, May 2001.

[8] C.-S. Chang, D.-S. Lee and C.-M. Lien, "Load balanced Birkhoff-von Neumann switches, Part II: multi-stage buffering," *Computer Comm.*, Vol. 25, pp. 623-634, 2002.

[9] R. Cole, K. Ost and S. Schirra, "Edge-Coloring Bipartite Multigraphs in O(E log D) Time," *Combinatorica*, vol. 21, pp. 5-12, 2001.

[10] J. Gripp *et al.*, "Demonstration of a 1.2 Tb/s optical packet switch fabric," *27th European Conf. on Optical Comm. (ECOC) '01*, Vol. 6, Amsterdam, The Netherlands, Oct. 2001.

[11] M.A. Marsan, A. Bianco, E. Leonardi, F. Neri and A. Nucci, "Multihop packet scheduling in WDM/TDM networks with nonnegligible transceiver tuning times," in *IEEE Transactions on Communications,* Vol. 48, No. 4, pp. 692–703, April 2000.

[12] G.P. Pieris and G.H. Sasaki, "Scheduling transmissions in WDM broadcast and select networks," in *IEEE/ACM Transactions on Networking,* Vol. 2, No. 2, pp. 105–110, 1994.

[13] G.N. Rouskas, V. Sivaraman, "Packet scheduling in broadcast WDM networks with arbitrary transceiver tuning latencies," in *IEEE/ACM Transactions on Networking,* Vol. 5, No. 3, pp. 359–370, 1997.

[14] A. Schrijver, "Bipartite edge-coloring in O(Δm) time," *SIAM J. Comput.*, vol. 28, pp. 841-846, 1999.

[15] A. Schrijver, "A course in combinatorial optimization", available on http://www.cwi.nl/~lex/files/dict.ps, Feb. 2003.

[16] J. von Neumann, "A certain zero-sum two-person game equivalent to the optimal assignment problem," *Contributions to the Theory of Games*, vol. 2, pp. 5-12, Princeton University Press, Princeton, NJ, 1953.

[17] T. Weller and B. Hajek, "Scheduling nonuniform traffic in a packet-switching system with small propagation delay," in *IEEE/ACM Transactions on Networking,* Vol.5, No. 6, pp. 813–823, 1997.

[18] I.M. White, M.S. Rogge, K. Shrikande, and L.G. Kazovsky, "Design of a control-channel-based media-access-control protocol for HORNET," *Journal of Optical Networking,* Vol. 1, No. 12, pp. 460-472, December 2002.

[19] M. Zirngibl, "Multifrequency lasers and applications in WDM networks," in *IEEE Communications Magazine,* Vol. 36, No. 12, pp. 39–41, December 1998.