Scheduling Multirate Sessions in Time Division Multiplexed Wavelength-Routing Networks

Suresh Subramaniam, *Member, IEEE*, Eric J. Harder, and Hyeong-Ah Choi, *Member, IEEE*

***Abstract*—We consider multiwavelength wavelength-routing networks operating in circuit-switched mode. Wavelength utilization is poor in such networks if sessions require only a fraction of a wavelength's capacity. An all-optical approach to improve wavelength utilization is to use time division multiplexing (TDM) on each wavelength, and switch time slots and wavelengths. In this paper, we address the off-line multirate session scheduling problem, i.e., the problem of assigning time slots and wavelengths to a given static set of multirate sessions, in ring topologies. Given a set of sessions and their relative rates, our objective is to maximize network throughput. This objective translates to the problem of minimizing the maximum length of a TDM frame over all wavelengths. We first show that the off-line single-rate session scheduling problem is equivalent to the off-line wavelength assignment problem, and hence obtain bounds on frame length. We then present scheduling algorithms with provable worst-case bounds on frame length for multirate session scheduling.**

***Index Terms*—Circuit switching, scheduling, time division multiplexing, wavelength routing.**

I. INTRODUCTION

WAVELENGTH ROUTING, together with wavelength division multiplexing (WDM), is a promising information transport mechanism for future all-optical networks. In a wavelength-routing network, network nodes route signals based on their wavelength and point of origin. Current optical technology limits the number of available wavelengths to a few tens, while allowing high transmission rates on each wavelength (e.g., 2.5 Gb/s).

While WDM has enabled more efficient access to optical fiber's vast bandwidth in comparison to non-WDM systems, fiber capacity could be greatly underutilized if wavelengths are underutilized. Consider a wavelength-routing network carrying circuit-switched traffic. Assigning one wavelength to each session could lead to poor wavelength utilization, since it is unlikely that every session needs the full bandwidth of a wavelength. Wavelength utilization can be improved by multiplexing many sessions on each wavelength.

An all-optical approach to improving wavelength utilization is to enable sessions to share a wavelength's bandwidth through a time division multiple access scheme [2]. In such a network, which we call a *time division multiplexed (TDM) wavelength routing network*, time is divided into frames of slots on each wavelength, and time slots and wavelengths are assigned to each session. The routing nodes (also called as wavelength routing switches) are capable of routing wavelengths as well as time slots, i.e., the routing patterns of nodes are configurable on a time slot basis. A TDM wavelength routing network operating in circuit-switched mode offers the benefits of transparency within the time slots assigned to a session.

In this paper, we consider a TDM wavelength routing network supporting *multirate circuit-switched sessions*. A session request is assigned one or more time slots (which are distributed over one or more wavelengths) forming an optical circuit, for the duration of the session. The rate of a session is proportional to the number of slots assigned to it, with one slot corresponding to a minimum rate session. The session transmits data in its assigned slots in each frame. At the end of each time slot, the wavelength routing nodes are reconfigured to reflect the routing pattern for the next time slot. The main challenge in the implementation of a TDM wavelength routing network is the quick reconfiguration of routing nodes. In a circuit-switched network, slots are assigned during session establishment, and therefore the cycle of switching patterns for each node is known once the sessions are established. The routing nodes can be *programmed* to follow this cycle of patterns for each frame. Allowing guard times between time slots and increasing the durations of slots can be used to mitigate the effects of slow reconfiguration times [3]. Note that because time on each wavelength is slotted, all-optical operation requires the global synchronization of the routing nodes at time slot boundaries. The feasibility of such a network has been demonstrated in the metropolitan area by the All Optical Network testbed [4]. In [4], a TDM wavelength routing network with tunable transceivers and a nonreconfigurable wavelength router is reported. Addressing was achieved in that network by launching the data on an appropriate wavelength. In the so-called B-service of the AON testbed, each frame is 250 μs long and consists of 128 slots.

We assume a static traffic model in this paper. The basic problem we address is the scheduling of multirate session requests. That is, given a set of sessions and their relative rates, assign time slots (and corresponding wavelengths) to each session so that an appropriate objective function is optimized. In this paper, our objective is to maximize the network throughput, while accommodating all session requests and respecting their relative rates. Our motivation for choosing this objective function and traffic model is the following. Given long-term traffic demands of sessions, it is reasonable to provision the network bandwidth in a fair manner (i.e., respect the relative rates of ses-

Manuscript received October 25, 1999; revised May 15, 2000. This work was supported in part by the NSA under LUCITE Contract MDA904-98-C-A081, and by the NSF under Grants ANI-9973098 and ANI-9973111. An earlier version of this paper appeared in *Proc. GLOBECOM*, Dec. 1999.

S. Subramaniam is with the Department of Electrical and Computer Engineering, George Washington University, Washington, DC 20052 USA (e-mail: suresh@seas.gwu.edu).

E. J. Harder and H.-A. Choi are with the Department of Computer Science, George Washington University, Washington, DC 20052 USA (e-mail: eharder@seas.gwu.edu; choi@seas.gwu.edu).

Publisher Item Identifier S 0733-8716(00)09000-4.

sions) and simultaneously maximize bandwidth utilization (network throughput). Moreover, assignment algorithms and performance analysis for a static traffic model can provide insights into the design and performance of algorithms for the dynamic traffic case. In this paper, we consider the ring topology for its practical suitability in metropolitan area networks.

Prior work on TDM wavelength routing all-optical networks has appeared in [4]–[7]. As mentioned earlier, a testbed that demonstrates the concept has been reported in [4]. In [5], the blocking probabilities of single-rate sessions (i.e., all sessions require one slot in a fixed frame) under dynamic traffic, with a random assignment of time slots and wavelengths, are obtained using analysis and simulations. The effects of wavelength converters and time slot interchangers on blocking performance were studied using a simple traffic model. The scheduling of all-to-all traffic (every node has a message to send to every other node) in ring topologies was studied in [6]. A fast, greedy, slot scheduling heuristic for transmitting packets is presented in [7].

The rest of the paper is organized as follows. The network model is presented in Section II. Scheduling algorithms for maximizing throughput under two different scheduling constraints are presented in Sections III and IV. The paper is concluded in Section V.

II. Network Model

The network topology is modeled by an undirected graph G with N nodes. Each edge corresponds to two unidirectional fibers. The sessions are undirected as they are assumed to be full-duplex and to use the same time slots and wavelengths in both directions. Each fiber supports W wavelengths (wavelength $0, \cdots, W-1$), and time slots on each wavelength are numbered starting from 0. We assume that there is an array of W transceivers at each port of a node, one tuned to each wavelength. Wavelength converters and time slot interchangers are assumed to be unavailable.

F is the set of sessions and $f_i \in F$ is the ith session. We assume that the rates are integral multiples of a basic rate (corresponding to one slot per frame on one wavelength).[1] Accordingly, each f_i is described by the ordered pair $\langle p_i, t_i \rangle$, where p_i is the path and t_i is the number of slots for session f_i.

A scheduling algorithm produces a schedule $\mathcal{S}$, i.e., assigns time slots and wavelengths to all sessions f_i. The frame length under schedule $\mathcal{S}$ is denoted as $T(\mathcal{S})$, and is defined to be the maximum number of slots assigned on any link over all wavelengths. It is easily seen that the network throughput under schedule $\mathcal{S}$ is $\rho(\mathcal{S}) = \sum_{i=1}^{|F|} (t_i \cdot C/T(\mathcal{S}))$, where C is the bit-rate per wavelength. Hence, our objective is to minimize $T(\mathcal{S})$. Note that the *duration* of the slots is not relevant here, and may be adjusted suitably. We now formally define the multirate sessions scheduling problem (MSSP).

Multirate Session Scheduling Problem (MSSP): Given an undirected graph G corresponding to a ring network with W wavelengths per link, and a set of sessions F where session f_i is described by its path and time slot requirement $\langle p_i, t_i \rangle$, obtain a schedule $\mathcal{S}$ such that the frame length $T(\mathcal{S})$ is minimized, under the given scheduling constraints.

[1]Equivalently, we assume that all the rates are rational multiples of the lowest rate session.

We consider two different scheduling constraints: 1) contiguous assignment of all slots of a session on one wavelength, and 2) noncontiguous assignment of a session's slots on possibly multiple wavelengths. Even though we do not explicitly consider switch reconfiguration times in this paper, restricting the slots of a session to be contiguous may relax the switching speed requirements on the wavelength routing switches. Therefore, the effect of the first constraint on frame length would be interesting to analyze. In the second case, we do not restrict the slots of a session to be contiguous or on a single wavelength. Note that if more than one wavelength is used for a session, then information on that session may need to be reordered by a higher layer entity at the destination. In both cases, no two sessions may be assigned the same time slot on the same wavelength if their paths share a link. The MSSP under the contiguous slot assignment constraint is examined in Section III, and the problem under the noncontiguous slot assignment constraint is studied in Section IV.

The following notation will be used throughout the paper.

1) $P = \{p_i | 1 \leq i \leq |F|\}$ is the set of session paths.
2) $P(e) = \{p_i \in P |$ path p_i contains link $e\}$ is the set of paths that contain link e.
3) $t(e) = \sum_{\{i | p_i \in P(e)\}} t_i$ is the total number of slots required by all sessions crossing link e.
4) $T_{\max} = \max\{t(e) | e \in E(G)\}$, where $E(G)$ is the set of edges in G, is the maximum number of slots required, over all links.
5) Let $\mathcal{S}_{\rm opt}$ denote an optimal schedule under the given constraints.

III. Contiguous Slot Assignment

In this section, we consider the MSSP under contiguous slot assignment. A schedule $\mathcal{S}$ is now defined as a set $\{\langle f_i, s_i, \lambda_i \rangle\colon\ 1 \leq i \leq |F|\}$, where s_i is the lowest indexed slot of session f_i in a frame and λ_i is the wavelength used for f_i. The frame length of $\mathcal{S}$ is given by $T(\mathcal{S}) = \max\{s_i + t_i | 1 \leq i \leq |F|\}$.

A. Single-Rate Sessions

We start by considering the case when all sessions are of the same rate, i.e., request the same number of slots. There is a close relationship between the single-rate session scheduling problem and the wavelength assignment problem (WAP) defined as follows. Given a graph G and a path set P, assign a wavelength to each path $p_i \in P$ such that the number of wavelengths is minimized subject to the constraint that no two paths share an edge and a wavelength.[2]

We derive upper and lower bounds on the frame length by transforming a solution to the WAP into a solution to the single-rate session scheduling problem (SSSP), and vice versa. There is no loss of generality in assuming that all sessions require one slot, i.e., $t_i = 1\, \forall i$.

Theorem 3.1 Lower Bound: Suppose we have an optimum schedule $\mathcal{S}_{\rm opt}$ for a graph G and path set P. Let χ be the number

[2]Note that W and t_i are not given in the WAP. In the WAP, each session requires a full wavelength, and the number of wavelengths in a solution to the WAP has no relation to W.

of wavelengths in a solution to the WAP for G and P. Then $T(\mathcal{S}_{\rm opt}) \geq \lceil(\chi/W)\rceil$.

Proof: Since all sessions require one slot, a set of sessions is scheduled to transmit in each slot of a frame of optimum length $T(\mathcal{S}_{\rm opt})$. Let F_{ij} be the set of sessions scheduled to transmit on slot i and wavelength j. Without loss of generality, we may assume that if a slot i on wavelength j is nonempty (i.e., if a session is scheduled on slot i and wavelength j), then all slots indexed $< i$, and slots indexed i on wavelengths indexed $< j$, are nonempty. Otherwise, we may simply schedule the sessions in F_{ij} in an empty slot without affecting optimality. Now, consider the following assignment for the WAP. The paths of all sessions in F_{ij} are assigned the wavelength $iW + j$. Note that this produces a valid wavelength assignment since $\mathcal{S}_{\rm opt}$ was a feasible schedule using W wavelengths. Therefore, we have $\chi \leq WT(\mathcal{S}_{\rm opt}) \Rightarrow T(\mathcal{S}_{\rm opt}) \geq (\chi/W) \Rightarrow T(\mathcal{S}_{\rm opt}) \geq \lceil(\chi/W)\rceil$, where the last implication follows from the fact that $T(\mathcal{S})$ is an integer for any schedule $\mathcal{S}$. □

Corollary 3.1: The SSSP is NP-complete.

Proof: When $W = 1$, the SSSP and the WAP are identical. The corollary follows from the fact that the WAP is NP-complete in rings [8].[3] □

Theorem 3.2 Upper Bound: Let K be the number of wavelengths in a given solution to the WAP for path set P in graph G. Then, a schedule $\mathcal{S}$ for the one-slot session scheduling problem for path set P in graph G given W wavelengths can be obtained, such that $T(\mathcal{S}) \leq \lceil(K/W)\rceil$.

Proof: Given a solution to the WAP with K wavelengths such that path p_j has wavelength i, $0 \leq i \leq K-1$, schedule session f_j at slot $\lfloor(i/W)\rfloor$ on wavelength i mod W. Since the given solution was a feasible wavelength assignment, all sessions assigned the same wavelength i in the WAP can be scheduled on the same slot $\lfloor(i/W)\rfloor$ and the same wavelength i mod W. Therefore, the resulting schedule is feasible. Since the solution to the WAP used K wavelengths, the highest slot index used in $\mathcal{S}$ is $\lfloor(K-1)/W\rfloor$, and the since the slots are numbered starting from 0, $T(\mathcal{S}) \leq \lfloor(K-1)/W\rfloor + 1 = \lceil(K/W)\rceil$. □

When all sessions require C slots for some integer $C > 1$, it is trivial to show that $T(\mathcal{S}) \leq \lceil K/W\rceil C$. This result will be used later in this section.

Using earlier results for the WAP, Theorems 3.1 and 3.2 directly lead to bounds on the frame length in the SSSP. For any topology, a simple lower bound is $T(\mathcal{S}_{\rm opt}) \geq \lceil T_{\max}/W\rceil$. Let $L_{\max}$ denote the maximum number of paths in P that cross any edge in G. When all sessions are one-slot sessions, note that $L_{\max} = T_{\max}$. Since a wavelength assignment algorithm for solving the WAP in ring networks with $\leq 2L_{\max} - 1$ wavelengths is known [9], a schedule $\mathcal{S}$ such that $T(\mathcal{S}) \leq 2T(\mathcal{S}_{\rm opt})$ can be constructed.

We next consider the more general case of multirate sessions.

B. Multirate Sessions

We now consider the case when the t_i's are not all equal. We first give a straightforward lower bound for the frame length applicable to any topology under contiguous slot scheduling.

[3]The WAP in rings is equivalent to coloring a circular arc graph.

Theorem 3.3 Lower Bound:

$$T(\mathcal{S}_{\rm opt}) \geq \max\left\{\max_{1\leq i\leq |F|} t_i, \left\lceil \frac{T_{\max}}{W}\right\rceil\right\}.$$

We now present suboptimal scheduling algorithms and upper bounds for frame length in the MSSP. Given a ring G and a session set F, we will find it useful to categorize the problem as one of two cases based on the following property of the path set P corresponding to F.

The Noncontainment (NC) Property: Consider two paths $p_i,\ p_j \in P$. Let $\{e_1(i), \cdots, e_{l_i}(i)\}$ be the ordered set of edges comprising p_i. Informally, p_i contains p_j if all the edges of p_j are in p_i, and neither the "leftmost" nor the "rightmost" edge of p_i is in p_j. A path set P is said to possess the *noncontainment* (NC) property if there is no pair of paths $p_i, p_j \in P$ such that p_i contains p_j. Note that if there are two paths between the same nodes, then neither contains the other.

Case 1— P Satisfies the NC Property: We first consider the case when the given path set P has the NC property, and derive an upper bound for frame length using the well-known *list-scheduling (LS)* [10] algorithm. The algorithm adapted to the MSSP under contiguous slot scheduling is shown in Fig. 1. Let the schedule produced by the LS algorithm be called $\mathcal{S}_{\rm LS}$.

Theorem 3.4 Upper Bound: $T(\mathcal{S}_{\rm LS}) \leq 3T(\mathcal{S}_{\rm opt})$.

Proof: Let f_m be a session assigned the last slot of a frame when the list-scheduling algorithm is applied. Recall that s_m is the smallest indexed slot assigned to session f_m. If $s_m = 0$, then there is nothing to prove.

Suppose $s_m > 0$. Let $\{e_1(m), e_2(m), \cdots, e_{l_m}(m)\}$ be the ordered set of edges comprising path p_m. Then, because of the NC property, the set of paths (other than p_m) sharing an edge with p_m can be partitioned into two sets as follows. Let $P_{\rm left}(m) = P(e_1(m)) - \{p_m\}$ and $P_{\rm right}(m) = P(e_{l_m}(m)) - \{p_m\} - P_{\rm left}(m)$. $P_{\rm left}(m)$ is the set of paths that contain the "leftmost" edge of p_m and $P_{\rm right}(m)$ is the set of paths that contain the "rightmost" edge of p_m and are not included in $P_{\rm left}(m)$. Because of the NC property and the ring topology, a path that does not contain either $e_1(m)$ or $e_{l_m}(m)$ does not share any edge with p_m.

Let $T_0^l(m)$ and $T_0^r(m)$ be the number of slots in a frame assigned to sessions with paths in $P_{\rm left}(m)$ and in $P_{\rm right}(m)$, respectively. By the LS algorithm, the session f_m is scheduled to start at the *smallest possible index* on an available wavelength. Therefore, during each slot $0, \cdots, s_m - 1$, sessions whose paths share edges with p_m must have been scheduled on each of the W wavelengths. Given this fact, we have

$$T_0^l(m) \leq \max\left\{ \max\{t_j | p_j \in P_{\rm left}(m)\}, \left\lceil \frac{\sum_{\{j|p_j\in P_{\rm left}(m)\}} t_j}{W} \right\rceil \right\}$$

Algorithm 1 LIST-SCHEDULING

Input: *A graph G and a session set $F = \{f_i\}$.*
Output: *A Schedule $\mathcal{S}$.*
/* *A is a set of ordered pairs $\langle k, s\rangle$ indicating that a currently scheduled session using wavelength k has $s-1$ as its highest slot index. Note that at a given time, A may have many ordered pairs with the same first element.* */

Set $\mathcal{S} = \emptyset$ and $A = \emptyset$.
for $0 \le k \le W-1$
  for $1 \le i \le |F|$
    if $f_i \in F$ can be assigned wavelength k using slots starting from 0,
      $\mathcal{S} = \mathcal{S} \cup \{\langle f_i, 0, k\rangle\}$;
      $F = F - \{f_i\}$;
      $A = A \cup \{\langle k, t_i\rangle\}$;
    endif
  endfor
endfor
while $(F \neq \emptyset)$
/* *Choose a wavelength k that is available for scheduling at the smallest indexed slot s and schedule as many sessions as possible using wavelength k starting from s.* */
  Let $\langle k, s\rangle \in A$ such that $s \le s'$ for any $\langle k', s'\rangle \in A$;
  $A = A - \{\langle k, s\rangle\}$;
  for $(1 \le i \le |F|)$
    if *$f_i \in F$ can be assigned wavelength k using slots starting from slot s,*
      $\mathcal{S} = \mathcal{S} \cup \{\langle f_i, s, k\rangle\}$;
      $A = A \cup \{\langle k, s + t_i\rangle\}$;
      $F = F - \{f_i\}$;
    endif
  endfor
endwhile
return $\mathcal{S}$.

Fig. 1. List-scheduling algorithm.

and

$$T_0^r(m) \le \max\left\{ \max\{t_j | p_j \in P_{\text{right}}(m)\}, \left\lceil \frac{\sum_{\{j|p_j \in P_{\text{right}}(m)\}} t_j}{W} \right\rceil \right\}.$$

Since

$$\sum_{\{j|p_j \in P_{\text{left}}(m)\}} t_j = t(e_1(m)) - t_m \le T_{\max}$$

and

$$\sum_{\{j|p_j \in P_{\text{right}}(m)\}} t_j \le t(e_{l_m}(m)) - t_m \le T_{\max}$$

using the lower bound of Theorem 3.3, we have $T_0^l(m) \le T(\mathcal{S}_{\text{opt}})$ and $T_0^r(m) \le T(\mathcal{S}_{\text{opt}})$. The theorem follows by noting that $s_m \le T_0^l(m) + T_0^r(m)$, $T(\mathcal{S}_{LS}) = s_m + t_m$, and $t_m \le T(\mathcal{S}_{\text{opt}})$. □

Remark: When $W = 1$, it is easy to show that $T(\mathcal{S}_{LS}) \le 2T(\mathcal{S}_{\text{opt}})$.

Case 2—P Does Not Satisfy the NC Property: In this case, the Proof of Theorem 3.4 can be generalized to yield an upper bound of $cT(\mathcal{S}_{\text{opt}})$ where c is $O(N)$. We present below an alternative scheduling algorithm that could provide a better upper bound for frame length when N is large.

An Alternative Scheduling Algorithm: Let l be an integer such that $2^{l-1} < (T_{\max}/W) \le 2^l$. This implies that $l = \lceil \log_2(T_{\max}/W)\rceil$. We divide sessions in F into $l+1$ groups such that a) $F_0 \overset{\text{def}}{=} \{f_j \in F | t_j \ge (T_{\max}/W)\}$, and b) $F_i \overset{\text{def}}{=} \{f_j \in F | (T_{\max}/2^i W) \le t_j < (T_{\max}/2^{i-1}W)\}$ for $1 \le i \le l$. Let P_i be the set of paths corresponding to the sessions in $F_i, 0 \le i \le l$. Let b_i $(0 \le i \le l)$ denote the maximum number of paths in P_i, using any edge e of G, i.e., $b_i \overset{\text{def}}{=} \max_{e \in E(G)} |P(e) \cap P_i|$. From the definition of F_i, we note that $b_i \le 2^i W$ for $0 \le i \le l$. Furthermore, let $C_0 \overset{\text{def}}{=} \max\{t_j | f_j \in F_0\}$ and $C_i \overset{\text{def}}{=} (T_{\max}/2^{i-1}W)$ for $1 \le i \le l$.

Now consider the following scheduling algorithm. The scheduling is done in $l+1$ disjoint rounds starting from round 0. In round $i, 0 \le i \le l$, sessions in F_i are scheduled using the scheduling algorithm for single-rate sessions (Section III-A) by assuming that all sessions in F_i require C_i slots. This is possible because $\max\{t_j | f_j \in F_i\} \leq C_i$. All W wavelengths are used in each round.

Before we derive an upper bound for the schedule length using this algorithm, we note that the lower bound of Theorem 3.3 can be rewritten as $T(\mathcal{S}_{\rm opt}) \geq \max\{C_0, \lceil T_{\max}/W \rceil\}$.

Theorem 3.5 Upper Bound: Let $T(\mathcal{S}_{\rm A})$ be the schedule length under the alternative scheduling algorithm. Then, $T(\mathcal{S}_{\rm A}) \leq 2(1 + 2\lceil \log_2 T(\mathcal{S}_{\rm opt}) \rceil) T(\mathcal{S}_{\rm opt})$.

Proof: Let χ_i denote the number of wavelengths in a solution to the WAP for path set P_i in G. From Theorem 3.2, the number of slots per frame, T_i $(0 \leq i \leq l)$, for scheduling sessions in F_i is no more than $\lceil \chi_i / W \rceil C_i$. From [9], we know that $\chi_i \leq 2b_i$. Since $b_0 \leq W$, we have $T_0 \leq 2C_0$. Moreover, since $b_i \leq 2^i W, 1 \leq i \leq l$, we have for $1 \leq i \leq l$,

$$T_i \leq \left\lceil \frac{\chi_i}{W} \right\rceil C_i \leq \left\lceil \frac{2b_i}{W} \right\rceil C_i \leq 2^{i+1} \frac{T_{\max}}{2^{i-1} W} = \frac{4T_{\max}}{W}.$$

Hence,

$$T(\mathcal{S}_{\rm A}) = \sum_{i=0}^{l} T_i \leq 2 \left[C_0 + \frac{2l T_{\max}}{W} \right] \leq 2(1 + 2l) T(\mathcal{S}_{\rm opt}).$$

□

IV. Noncontiguous Slot Assignment

We now investigate the MSSP when the slots of a session need not be contiguous or on a single wavelength. Note, however, that since wavelength converters and time-slot interchangers are absent, the assigned slot and wavelength indices have to be the same on all the links of a session's path. The NP-completeness of the MSSP under this constraint trivially follows from the NP-completeness of the SSSP[4] (Section III-A).

A solution approach to this problem is to replace each session f_i by t_i sessions requiring one slot each, and then applying the algorithm for SSSP. However, this is not a polynomial-time algorithm because the number of new sessions created is $\sum_{i=1}^{|F|} t_i$ which may not be bounded by any polynomial in $|F|$, N, or W. We now present an algorithm with time complexity $O(|F|(N + |F|))$ that yields the same bound as the approach mentioned above.

A schedule $\mathcal{S}$ is now defined as a set $\{\langle f_i, s_i^j, d_i^j, \Lambda_i^j \rangle : 1 \leq i \leq |F|, 1 \leq j \leq M_i\}$, where M_i is the number of contiguous sets of slots (defined as *blocks*) assigned to f_i by our schedule and s_i^j is the lowest indexed slot of the jth block assigned to f_i. d_i^j denotes the number of slots (on each of a set of wavelengths Λ_i^j) assigned in the jth block to f_i. The schedule length of $\mathcal{S}$ is then given by

$$T(\mathcal{S}) = \max \left\{ s_i^j + d_i^j \,\middle|\, \left\langle f_i, s_i^j, d_i^j, \Lambda_i^j \right\rangle \in \mathcal{S} \right\}.$$

The straightforward lower bound is as follows:

Theorem 4.1—Lower Bound: $T(\mathcal{S}_{\rm opt}) \geq \lceil T_{\max}/W \rceil$. Without loss of generality, we assume that $t(e) = T_{\max} \; \forall e \in E(G)$, and that[5] $T_{\max}/W = \lceil T_{\max}/W \rceil$. For any path $p_i \in P$, let $E(p_i)$ denote the set of edges used by p_i. For any path set $P' \subseteq P$, let $E(P')$ denote the set of edges used by at least one path in P'. Finally, let $Q(p_i)$ denote the set of paths in P that overlap with p_i (i.e., every path in $Q(p_i)$ shares at least one edge with p_i.). We start deriving the upper bound with a lemma.

Lemma 4.1: For a bus topology G, $\exists$ a subset $P_0 \subseteq P$ such that $|P(e) \cap P_0| = 1 \; \forall e \in E(G)$.

Proof: Let $\{e_i | 1 \leq i \leq N-1\}$ be the set of edges in $E(G)$ listed from left to right. We have to find P_0 such that every edge in G is crossed exactly once by some path (not necessarily the same path) in P_0. Such a set can be constructed in the following way.

Initially, let $P_0 = \emptyset$, $E^m = E(G)$, and $P^1 = P$. In each iteration, P^1 denotes the set of paths that have not yet been eliminated from consideration as elements of P_0. Consider the leftmost edge (e_1 at the first iteration) in E^m. Let p_{i_1} be an arbitrary path in P^1 using e_1. We then update P_0, E^m, and P^1 such that $P_0 = P_0 \cup \{p_{i_1}\}$, $E^m = E^m - E(p_{i_1})$, and $P^1 = P^1 - Q(p_{i_1})$. We repeatedly consider the leftmost edge in E^m and apply this procedure until E^m becomes empty.

To prove that the above procedure always produces a desired set, let us consider the ith iteration of the procedure for an arbitrary i. Assume that P_0, E^m, and P^1 have been properly updated before the ith iteration. Let e_k be the leftmost edge in E^m. Then, note that there exists at least one path in P^1 that uses e_k [i.e., $P(e_k) \cap P^1 \neq \emptyset$]. If not, $P(e_k) \subseteq Q(p_{i'})$, where $p_{i'}$ is the path selected at the $(i-1)$th iteration. Since $e_k \notin E(p_{i'})$, this implies that $t(e_k) < T_{\max}$; a contradiction. □

Our schedule for rings is based on an optimal schedule for bus topologies. We now present this schedule which uses Lemma 4.1. Loosely speaking, our algorithm runs several iterations and finishes the assignment of slots for at least one session in each iteration. The assignment is done in a greedy manner by first completing the assignment of the lowest indexed possible slot over all wavelengths, before proceeding to the next higher indexed slot. We formalize this below.

Initially, set $s = 0, k = 0$, and $\mathcal{S} = \emptyset$. Our algorithm will run multiple iterations, and a set of type P_0 is computed as discussed in Lemma 4.1 in each iteration. The jth iteration of the algorithm is outlined in the following. Let $P_0^j \subseteq P$ be the set of type P_0, and $b_j = \min\{t_i | p_i \in P_0^j\}$. b_j is the number of slots (counting slots with the same index but over different wavelengths as different slots) that will be assigned in this iteration, and hence all sessions $p_i \in P_0^j$ with $t_i = b_j$ will have their assignments completed at the end of this iteration.

Let $x_j = \min(b_j, W - k)$, and $y_j = \max(0, b_j - x_j)$. Set $\mathcal{S} = \mathcal{S} \cup \{\langle f_i, s, 1, \{k, k+1, \cdots, k + x_j - 1\} \rangle | p_i \in P_0^j\}$, and $k = (k + x_j) \bmod W$. If $k = 0$, set $s = s + 1$. Define $d_1^j = \lfloor y_j / W \rfloor$, and $d_2^j = y_j \bmod W$. If $d_1^j > 0$, set $\mathcal{S} = \mathcal{S} \cup \{\langle f_i, s, d_1^j, \{0, 1, \cdots, W-1\} \rangle | p_i \in P_0^j\}$, and set $s = s + d_1^j$. Furthermore, if $d_2^j > 0$, set $\mathcal{S} = \mathcal{S} \cup \{\langle f_i, s, 1, \{0, 1, \cdots, d_2^j - 1\} \rangle | p_i \in P_0^j\}$. Finally, set $k = k + d_2^j$, and $t_i = t_i - x_j - W d_1^j - d_2^j$ for all sessions f_i such that $p_i \in P_0^j$.

[4]Note that the contiguous and noncontiguous slot assignment constraints are identical if all sessions require one slot.

[5]If $t(e)/W < \lceil T_{\max}/W \rceil$, for some $e \in E(G)$, a one-hop dummy session can be added to make $t(e)/W = \lceil T_{\max}/W \rceil$. For the rest of this section, P refers to the set of paths including any dummy paths.

At the end of the iteration, we update F and P such that $F = \{f_i \in F | t_i > 0\}$, and $P = \{p_i | f_i \in F\}$. The above procedure is repeated until the schedule is complete. Observe that the algorithm "fills up" all slots with a given index before proceeding to assign slots with a higher index. Hence, the schedule is optimal.

Since the number of dummy sessions that have to be added is $\leq N$, P_0 in Lemma 4.1 can be computed in $O(N + |F|)$ time. We note that at least one nondummy session is completed in each iteration, leading to a time complexity of $O(|F|(N+|F|))$ for the scheduling algorithm.

Theorem 4.2—Upper Bound: There is an algorithm $\mathcal{S}$ that produces a noncontiguous schedule such that $T(\mathcal{S}) \leq 2\lceil T_{\max}/W \rceil$ for rings.

Proof: Choose any edge e and consider the set of sessions $F' = \{f_i | p_i \in P(e)\}$. Then the paths for the sessions in $F - F'$ do not contain e. Therefore, by using the schedule for the bus topology, sessions in $F - F'$ can be scheduled using no more than $\lceil T_{\max}/W \rceil$ slots. Since the sessions in F' can all be scheduled using no more than $\lceil T_{\max}/W \rceil$ slots, the theorem follows. $\square$

V. Conclusions and Future Work

In this paper, we considered the problem of scheduling multirate sessions in time-division multiplexed wavelength routing rings so that network throughput is maximized. Maximizing network throughput while respecting relative session rates is equivalent to minimizing the frame length. Given a static set of sessions with slot requirements, we presented scheduling algorithms with provable worst-case bounds on frame length.

Bounds on frame length were presented under two scheduling constraints. When the slots of a session have to be contiguous and on a single wavelength, we proved that the list scheduling algorithm achieves a frame size that is at most three times the optimal size. This result assumes that no session's path contains another session's path. When this assumption was not valid, we presented an algorithm that could provide a better upper bound than the list scheduling algorithm. On the other hand, when a session's slots need not be contiguous or on a single wavelength, then a scheduling algorithm that achieves twice the optimal frame size was presented. These results could be useful in providing insights into the more practical on-line scheduling and connection admission control. To the best of our knowledge, these are the first bounds for the multirate session scheduling problem. Simulation results (presented in [1]) indicate that the lower bounds are significantly tighter on average than the upper bounds.

The work in this paper can be extended in many directions. An obvious extension is to pursue better bounds. In this paper, we have not explicitly taken into account the switch reconfiguration time in developing the schedules. The design of schedules considering switching times, a limited number of transceivers, and transceiver tuning times remains to be studied. Another interesting problem to study is the on-line (dynamic) scheduling of transmissions. Finally, blocking schedules that guarantee a fixed rate to sessions and their performance are topics for future investigation.

References

[1] S. Subramaniam, E. J. Harder, and H.-A. Choi, "Scheduling multi-rate sessions in TDM wavelength-routing networks," in *Proc. GLOBECOM*, Dec. 1999, pp. 1466–1472.

[2] M. Sabry and J. Midwinter, "Toward an optical ether," *Proc. Inst. Elect. Eng. J.*, vol. 141, no. 5, pp. 327–335, Oct. 1994.

[3] H. S. Hinton, *An Introduction to Photonic Switching Fabrics*. New York: Plenum, 1993.

[4] I. P. Kaminow *et al.*, "A wideband all-optical WDM network," *IEEE J. Select. Areas Commun.*, vol. 14, pp. 780–799, June 1996.

[5] J. Yates, J. Lacey, and D. Everitt, "Blocking in multiwavelength TDM networks," in *Proc. 4th Int. Conf. Telecommun. Syst., Modeling, Anal.*, Mar. 1996, pp. 535–541.

[6] X. Zhang and C. Qiao, "On scheduling all-to-all personalized connections and cost-effective designs in WDM rings," *IEEE/ACM Trans. Networking*, vol. 7, pp. 435–445, June 1999.

[7] A. C. Kam, K.-Y. Siu, R. A. Barry, and E. A. Swanson, "Toward best-effort services over WDM networks with fair access and minimum bandwidth guarantees," *IEEE J. Select. Areas Commun.*, vol. 16, pp. 1024–1039, Sept. 1998.

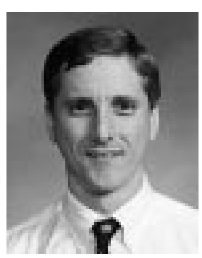
[8] M. Garey, D. Johnson, G. Miller, and C. Papadimitriou, "The complexity of coloring circular arcs and chords," *SIAM J. Disc. Math.*, vol. 1, no. 2, pp. 216–227, June 1980.

[9] A. Tucker, "Coloring a family of circular arcs," *SIAM J. Appl. Math.*, vol. 29, no. 3, pp. 493–502, Nov. 1975.

[10] E. J. Coffman and P. J. Denning, *Operation Systems Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1973.

Suresh Subramaniam (S'95–M'97) received the B.E. degree in electronics and communication engineering from Anna University, Madras, India, in 1988, the M.S.E.E. degree from Tulane University, New Orleans, LA, in 1993, and the Ph.D. degree in electrical engineering from the University of Washington, Seattle, in 1997.

He was a Hardware Engineer in the Computer Division, HCL, Ltd., Madras, from 1988 to 1990, and a Research Assistant in the Electrical Communication Engineering Department, Indian Institute of Science, Bangalore, from 1990 to 1991. He was a Staff Member in the Optical Communications Technology Group at the Lincoln Laboratory, Massachusetts Institute of Technology, Lexington, in 1996. He is an Assistant Professor in the Department of Electrical and Computer Engineering, George Washington University, Washington, DC, where he has been since September, 1997. His teaching and research interests are in the areas of optical and wireless networks. He is a coeditor of the book *Optical WDM Networks—Principles and Practice* (Boston, MA: Kluwer, 2000).



Eric J. Harder received the B.S. degree in physics from the College of William and Mary, Williamsburg, VA, the B.S. degree in mathematics from the University of Maryland, College Park, and the M.S. and Ph.D. degrees in computer science from George Washington University, Washington, DC.

He is a Senior Computer Scientist at the National Security Agency. His research interests include network control and survivability, and the analysis of algorithms, control, and communication systems.

Hyeong-Ah Choi (M'82) received the B.A. and M.S. degrees from Seoul National University, Seoul, Korea, in 1980 and 1982, respectively, and the Ph.D. degree in computer science from Northwestern University, Evanston, IL in 1986.

He is a Professor of computer science at George Washington University (GWU), Washington, DC. Before joining GWU in 1988, she was an Assistant Professor of computer science at Michigan State University, East Lansing, for two years. Her research interests include optical networks, parallel and distributed computing, network survivability, algorithms, and graph theory. Her research has been supported by the National Science Foundation, National Security Agency, National Institute of Standards and Technology, and DARPA.