LOAD BALANCING IN ALL-OPTICAL OVERLAID-STAR TDM NETWORKS

Jun Zheng, Cheng Peng, Gregor v. Bochmann, and Trevor J. Hall
School of Information Technology and Engineering
University of Ottawa
Ottawa, Ontario K1N 6N5, Canada
Email: {junzheng, cpeng, bochmann, thall}@site.uottawa.ca

Abstract **Load balancing is an effective solution to relieving network congestion and achieving good network performance. This paper investigates routing strategies for load balancing in all-optical overlaid-star TDM networks. A random routing strategy and a least-congested-path routing strategy are first presented, based on which a weighted-least-congested-path routing strategy is then proposed. The proposed strategy takes into account both load balancing and end-to-end delay in path selection, and thus can achieve better delay performance while maintaining the same blocking performance under low traffic load as compared with the other strategies. The performance of the routing strategies is evaluated through simulation results.**

Index Terms—Load balancing, overlaid star topology, TDM, optical network

I. INTRODUCTION

All-optical overlaid-star TDM networks are a class of all-optical networks that employ an overlaid-star topology and use time division multiplexing (TDM) for data transmission. This class of networks features the ability to dynamically allocate bandwidth on demand at a fine granularity, and the concentration of control and routing functionality at the electronic edge nodes that surround the optical core [1]. The architecture of such networks consists of a number of edge nodes interconnected via several core nodes in an overlaid-star topology, as shown in Figure 1. The overlaid-star topology provides robustness in the event of a network failure and at the same time relieves potential network congestion. To achieve good network performance, traffic load should be distributed over the multiple stars in a balanced manner. For this reason, load balancing becomes a critical issue in such networks. While this issue has widely been studied for mesh optical networks, it has not been studied for all-optical overlaid-star TDM networks. The objective of this work is to seek an efficient routing strategy for load balancing in such networks. To this end, a *random* routing strategy and a *least-congested-path* routing strategy are first presented. Based on this, a *weighted-least-congested-path* routing strategy is then proposed, which takes into account both load balancing and end-to-end delay in path selection. The purpose is to improve the delay performance while maintaining the blocking performance of the network.

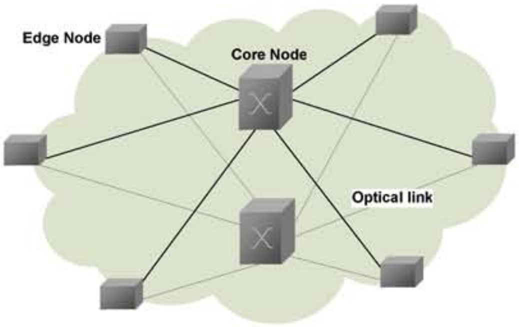

Figure 1. Overlaid-star topology.

II. BACKGROUND

In this section, we briefly describe the network architecture and review related work on load balancing.

A. Network Architecture

An all-optical overlaid-star TDM network consists of a number of edge nodes interconnected via several central core nodes in an overlaid star topology, as shown in Figure 1. Each edge node is connected to a core node by a couple of fibers, one for transmission in each direction. An edge node is a hybrid electro-optical device that serves as an interface between the optical network and an electronic external networks, based on IP, MPLS, or ATM. A core node employs an all-optical space switch that can switch an input wavelength on an input port to an output port, making data paths inside the core node purely optical and transparent.

The network uses time division multiplexing (TDM) for data transmission. Each fiber supports multiple wavelengths. Each wavelength is divided into a series of frames that consist of a fixed number of timeslots. The control of the network is performed in the electronic domain. Each core switch has an associated electronic controller that performs timeslot allocation, switch configuration, and other control functions. The control messages are exchanged between edge nodes and core nodes out-of-band over a dedicated control timeslot on a particular wavelength of each fiber or over a dedicated wavelength of each fiber. There is one control timeslot per frame in either direction.

B. *Related Work*

Load balancing has been widely studied in optical networks. The purpose of load balancing is to relieve network congestion and improve network performance [2]. In a packet/burst-switched network, load balancing can reduce packet/burst delay and loss, and thus improve quality of service (QoS). In a circuit-switched network, it can reduce blocking probability and thus accommodate more connections.

In this work, we consider circuit-switched networks. In this context, Brunato et al. proposed a load-balancing algorithm called Reverse Subtree Neighborhood Exploration (RSNE) for dynamic lightpath establishment in wavelength-routed networks [3]. It is based on IP-like routing and a local searching mechanism, and allows an incremental implementation where local searching steps are continuously performed as traffic conditions change. In [4], Hse et al. investigated adaptive routing algorithms for wavelength-routed networks, including the least-loaded path strategy and the proposed weighted shortest path strategy. The former can balance traffic load well among all links. The latter can minimize resource cost while maintaining traffic load among all links as balanced as possible. In [5], Narula-Tam et al. proposed algorithms for virtual topology reconfiguration in wavelength-routed networks, which consider load balancing in adapting to rapid changes of traffic patterns. The proposed algorithm achieves load balancing by minimizing the maximum link load in the network.

III. ROUTING STRATEGIES FOR LOAD BALANCING

In this section, we first present two routing strategies for load balancing: *random routing* and *least-congested-path routing*, and then propose a new routing strategy that takes into account both load balancing and end-to-end delay in path selection.

The overlaid star network can be modeled as a three-dimensional directed graph $G = (V_1, V_2, E)$, where V_1 and V_2 denotes two different sets of vertices and E stands for the set of directed edges. Each vertex v_i $(i = 0, \cdots, N-1)$ in V_1 corresponds to an edge node and each vertex v_k $(k = 0, \cdots, M-1)$ in V_2 corresponds to a core node, where N is the number of edge nodes and M is the number of core nodes in the network. Each edge e_{ik} (or e_{ki}) in E corresponds to a fiber link between edge node i and core node k (or between core node k and edge node i). There are L timeslots in each frame on each fiber link, which are denoted by $S = \{s_1, s_2, \cdots, s_L\}$.

We consider a dynamic traffic model in which calls (or connection requests) arrive at each edge node dynamically and the bandwidth demand of each call can be one or more timeslots. A connection must first be established between a pair of edge nodes before data is transferred. In such a network, there exists a set of fixed M paths between a pair of edge nodes, each passing through one of the core nodes, as shown in Figure 2. The set of paths between edge node i and edge node j are denoted by $P_{ij} = \{p_{ij}^k; i, j = 0, \cdots, N; k = 0, \cdots, M\}$, where $p_{ij}^k = \{e_{ik}, e_{kj}\}$ denotes the path passing through core node k and consisting of edge e_{ik} and edge e_{kj}. To establish a connection for a call, the source node must first select one of the *M* paths to the destination. The traditional shortest-path strategy may result in a situation where some of the links are overloaded while the others are underloaded, which would affect the performance of the network. To achieve load balancing on each link, each source node should distribute traffic load among the *M* paths to each destination as evenly as possible.

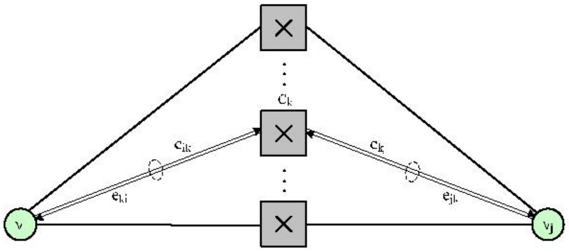

Figure 2. A set of two-hop paths.

A. *Random Routing Strategy*

In the *random* routing strategy, for each call, the source node randomly selects one of the *M* paths to the destination in a uniform manner and then uses a signaling protocol to establish a connection for the call. If the connection is not established successfully, the request is dropped. This strategy is simple to implement but it does not take into account link state information and thus may not be able to achieve the best performance. A variant of this strategy is called *random-with-retrying*, which introduces retrying in path selection. In the event of an unsuccessful establishment, the source node randomly selects another path among the rest paths, which would significantly improve the blocking performance of the network. The request will be dropped if all paths are tried with no success.

B. *Least-Congested-Path Routing Strategy*

In the *least-congested-path* routing strategy, path selection is based on the current timeslot usage on each link of a path. For each call, the source node selects the least congested path among all *M* paths to the destination, making the timeslot usage on each link more balanced and the network performance improved. The congestion of a path is defined as the number of timeslots available on the most congested link of the path. The congestion of a link is measured in terms of the number of timeslots available on the link. The fewer the number of available timeslots, the more congested the link. For a call from node i to node j, the source node selects a path $p_{ij}^k = \{e_{ik}, e_{kj}\}$ that passes through core node k and satisfies

$$\underset{k}{Max}\{Min[(L-l_{ik}),(L-l_{kj})]\},\ k=0,1,\cdots,M-1$$

where l_{ik} and l_{kj} are the number of timeslots already used on e_{ik} and e_{kj}, respectively.

To support this strategy, each edge node must maintain the state (or congestion) information on each link. This information should be advertised and updated by each core node periodically using a signaling protocol, typically one time each frame. The link state information should contain the timeslot usage in each frame. Note that the link state information used to make a path selection by the source may be outdated because of the propagation delay on each link, which would affect network performance.

Compared with the *random* strategy, this strategy is more complex to implement because it requires the core nodes to advertise and update the link state information periodically and the source nodes to compute a path based on the link state information it maintains.

C. *Weighted-Least-Congested-Path Routing Strategy*

The *least-congested-path* routing strategy can effectively balance the traffic load on each link and thus improve the blocking performance of the network. For a particular call, however, it may select a longer path instead of an available shorter path, which would increase the end-to-end delay of the connection. Actually, load balancing is unnecessary under low traffic load. In this case, there is no congestion in the network. The source node can select the shortest available path for each call, which would improve the delay performance of the network while not affecting the blocking performance. Based on this argument, we propose a *weighted-least-congested-path* routing strategy that takes into account both load balancing and end-to-end delay in path selection. For a call from node *i* to node *j*, the source node selects a path $p_{ij}^{k}=\{e_{ik},e_{kj}\}$ that passes through core node *k* and satisfies

$$\begin{cases}\underset{k}{Min}\{d_{ik}+d_{kj}\} & if\ \rho\le 0.5/M\\ \underset{k}{Max}\{Min[(L-l_{ik}),(L-l_{kj})]\} & otherwise\end{cases}$$

where d_{ik} and d_{kj} are the link distances of e_{ik} and e_{ik}, respectively, and ρ is offered traffic load. The reason to choose *0.5* is based on the observation that there is nearly no blocking when traffic load is less than *0.5* and all traffic takes the shortest paths.

IV. PERFORMANCE EVALUATION

In this section, we evaluate the performance of the routing strategies discussed in Section III through simulation results.

A. *Simulation Model*

We consider a network with eight edge nodes (*N=8*) and two core nodes (*M=2*). The node layout and link distance (in *km*) are given in Table I. The call arrival process is Poisson and the duration of each call is exponentially distributed. The mean arrival rate is λ and the mean holding time is $1/\mu$. For each call, the destination node is uniformly distributed. The bandwidth demand of each call is one timeslot.

TABLE I. NODE LAYOUT AND LINK DISTANCE

Cores **Distance** **Edges**	Toronto	Montreal
Quebec	792	248
Montreal	564	0
Ottawa	396	195
Toronto	0	546
Waterloo	105	651
London	186	731
Hamilton	68	610
Windsor	366	910

B. *Simulation Results*

Figures 3, 4, and 5 show the average blocking probability, end-to-end delay, and set-up time, respectively, with different routing strategies. The results are obtained with a frame size equal to *100* timeslots and the mean holding time of each call equal to *100* timeslots.

In Figure 3, one can observe that all the strategies achieve good blocking performance when traffic load is low. This means that the *weighted-least-congested-path* routing strategy can achieve zero blocking probability even when the shortest-path routing is used under low traffic load. The *least-congested-path* routing strategy and the *weighted-least-congested-path* routing strategy have the same blocking performance. The *random* routing strategy shows the worst performance because it does not take into account current link state information. The *random-with-retrying* strategy shows the best performance because of retrying it has introduced. Note that *retrying* can also be applied to the other strategies to further reduce the blocking probability.

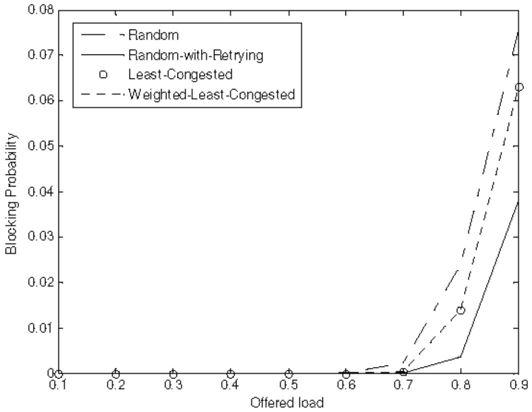

Figure 3. Blocking probability versus traffic load.

In Figure 4, it is observed that the *weighted-least-congested-path* routing strategy achieves better end-to-end delay performance than the other strategies when traffic load is lower than *0.25* because the shortest paths are selected instead of the least-congested paths. The improvement depends on the propagation delay of the shortest paths.

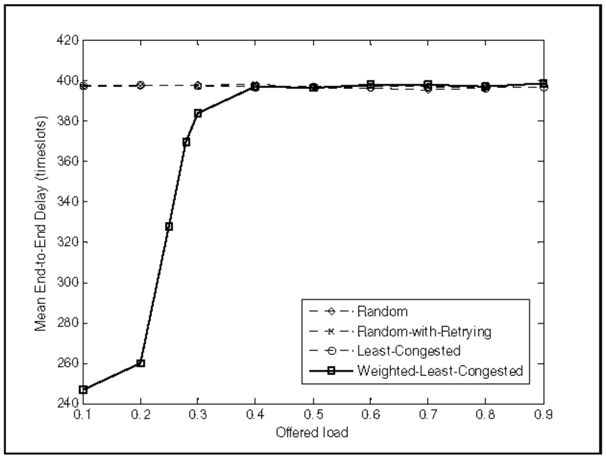

Figure 4. Mean end-to-end delay versus traffic load.

In Figure 5, it is observed that the *weighted-least-congested-path* routing strategy achieves better set-up delay performance than the other strategies when traffic load is lower than 0.25. This is because the shortest paths are selected and it takes the signaling protocol less time to establish a connection. Although the *random-with-retrying* strategy improves the blocking performance when traffic load is larger than 0.7, it results in a larger connection set-up time.

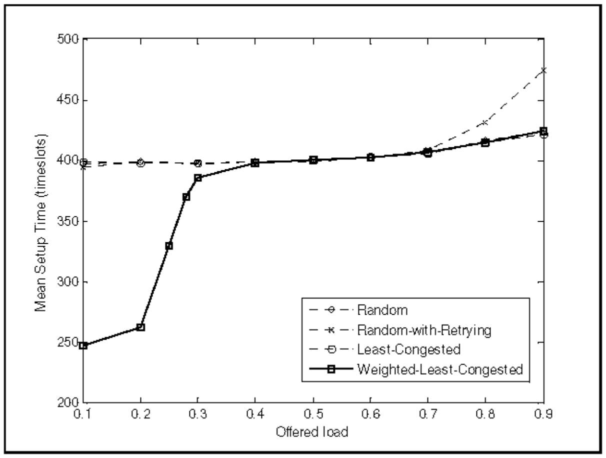

Figure 5. Set-up time versus traffic load.

V. CONCLUSIONS

In this paper, we investigated routing strategies for load balancing in all-optical overlaid-star TDM networks. The *random* routing strategy and the *least-congested-path* routing strategy were first presented, and the *weighted-least-congested-path* routing strategy was then proposed to improve the delay performance under low traffic load. The simulation results show that the proposed *weighted-least-congested-path* routing strategy can significantly improve the delay performance under low traffic load while maintaining the same blocking probability as that of the other routing strategies. The *random-with-retrying* strategy has the best blocking performance but at the cost of a larger connection set-up time.

ACKNOWLEDGMENT

This work was supported by the Natural Sciences and Engineering Research Council (NSERC) of Canada and industrial and government partners, through the Agile All-Photonic Networks (AAPN) Research Network. Dr. Trevor J. Hall holds a Canada Research Chair in Photonic Network Technology at the University of Ottawa and is grateful to the Canada Research Chairs Program for their support.

REFERENCES

[1] G. v. Bochmann et al., "The agile all-Photonic Network: an architecture outline," Proc. of Queen's University Biennial Symposium on Communications, 2004, pp.217-218.

[2] J. Zheng and H. T. Mouftah, *Optical WDM Networks: Concepts and Design Principles*, Wiley-IEEE Press, New Jersey, Jul. 2004.

[3] M. Brunato, R. Battiti, and E. Salvadori, "Dynamic load balancing in WDM networks," Optical Networks Magazine, Vol. 4, No. 5, Sept./Oct. 2003, pp. 7-20.

[4] Ching-Fang Hsu, Te-Lung Liu, and Nen-Fu Huang, "Performance of adaptive routing strategies for wavelength-routed networks," Proc. of 2001 IEEE International Conference on Performance, Computing, and Communications, Apr. 2001, pp. 163-170.

[5] A. Narula-Tam and E. Modiano, "Dynamic load balancing in WDM packet networks with and without wavelength constraints," IEEE Journal on Selected Areas in Communications, Vol. 18, No. 10, Oct. 2000 pp. 1972 – 1979.