Latency in Verifying Connection Setup in Dynamic Optical Networks

Jane M. Simmons, *Fellow, IEEE*

***Abstract*—We consider dynamic applications that require new connections to be established with very low setup times. One challenge is determining when to initiate transmission, which can be based on a timing mechanism or based on receiving positive acknowledgment that the connection has been properly established. We investigate the latter strategy and analyze the added latency that is imposed by requiring verification of a newly established path. We show that with a distributed connection-setup architecture, verification potentially adds a negligible amount of latency, whereas with centralized operation, the added latency can be appreciable, especially in a backbone network.**

***Index Terms*—Centralized, connection setup, distributed, dynamic networks, latency, path verification, PCE, RSVP-TE.**

I. Introduction

OPTICAL transport networks have grown increasingly more dynamic as a means of using capacity more efficiently and responding to changes in the network state [1]-[5]. For example, dynamic networking can be used to rapidly deliver more bandwidth to congested areas of the network or to pro-actively migrate a connection to a new path in anticipation of a failure of the original path. In addition to these network-driven operations, end-user applications with low-latency setup requirements also are burgeoning [6]. For example, in a backbone network, applications such as large-scale distributed computing or data fusion from geographically distributed sites require connection setup on the order of 100 ms [7]. At the network edge, applications such as augmented reality require network response times on the order of 10 ms [8].

Connection establishment encompasses numerous steps including determining the new path, communicating the required setup to the relevant network nodes, and configuring the switches as well as other network equipment needed to support the new connection. Another more subtle component of the process is determining when the connection is successfully established and transmission can actually begin [9], [10]. This can be addressed with a timing mechanism, such that transmission starts after a pre-determined amount of time that is calculated based on estimates of the required times needed for the various connection establishment steps. Depending on the aggressiveness in setting the timer, this can result in minimal excess latency. However, one drawback with relying on a timer is that transmission may commence even though the connection has not been properly established. This could occur, for example, due to a switch being misconfigured (e.g., a switch can get stuck such that the desired new configuration is not achieved). This potentially results in an alternative undesirable end-to-end path, where data is delivered to the wrong destination.

Misdelivered data is a potential security risk. While it may be tempting to argue that encrypting the data mitigates this risk, it can be shown that important information can still be gleaned from encrypted data [11]; e.g., based on the amount of data, time of day, or other patterns in the traffic.

Thus, especially for mission-critical applications, it is worthwhile to consider schemes where a positive acknowledgment of the proper connection setup must be received by the source prior to commencing transmission. The tradeoff is how much latency is added by this requirement and is it compatible with applications with very demanding setup times.

The purpose of this letter is three-fold. First, it raises awareness of the verification aspect of connection setup, a facet that is often ignored in dynamic networking studies. Second, it investigates the delays imposed by requiring positive acknowledgment. This analysis is performed at a high level in order to focus on the delays inherent to the various schemes considered. Third, it specifically compares connection verification latencies in centralized (Section II) and distributed (Section III) schemes, in the context of both backbone and metro networks. Choosing a centralized or distributed architecture for connection setup is an important dichotomy in network design. We show that the potentially lower verification latency inherent to a distributed architecture, especially in a backbone network, may be significant enough to impact this decision. The discussion focuses on establishing wavelength-level connections in the optical layer; the conclusions extend more generally to other circuit-based layers of the network.

II. Verification in a Centralized Architecture

In this section, we assume a centralized architecture is employed for path calculation and setup. A representative example is shown in Fig. 1. It is assumed that a new connection request between Node A (the source) and Node Z (the destination) arrives at the source. The source sends the request to a centralized controller, typically located in a geographically strategic location. We generically refer to this centralized controller as the path computation element (PCE) [12]. (The analysis can be readily extended to a scenario where the PCE initiates the connection request [13].)

The PCE calculates the desired path from A to Z, the wavelength to be used on each link of the path, and the requisite reconfigurations that must occur along the path; e.g., the new cross-connect pattern in the optical switch at each node to create a path from source to destination. The

Manuscript received July 23, 2018.

J. M. Simmons is with Monarch Network Architects, Holmdel, NJ, USA.

optical switch is typically a reconfigurable optical add/drop multiplexer (ROADM). It is assumed that to minimize latency, the PCE sends the configuration instructions directly to each of the nodes along the path.

Let L_i be the one-way propagation time between the PCE and Node i, and let R_i be the one-way propagation time between the source and Node i. Let Node F be the node that is furthest from the PCE; i.e., it is the last node to receive the configuration instructions from the PCE. We assume that the propagation times are the same in either direction of a link. Furthermore, we assume a scenario with the control plane parallel to the data plane, with similar delays; this assumption only comes into play in the timer-based start scheme, not in the schemes based on acknowledgments.

We assume that the equipment configuration time, S, is the same at all nodes. (It is straightforward to extend the analysis if S is different at some nodes.) The configuration of the equipment commences as soon as the instruction from the PCE is received. In the analyses below, S is assumed to be 15 ms. This includes turning up or retuning optical transponders, reconfiguring the optical switch, and stabilization of the optical signal. The choice of 15 ms is based on discussions with switch vendors as to the delay that is achievable, assuming that optical transients are well managed [3], [4].

To reduce the number of variables to track, we assume that any processing or queuing delays, whether at the PCE or at the nodes, are included in L and R. We expect these delays to be relatively small (e.g., less than 1 ms [3]) under normal (non-failure, non-heavily-congested) conditions. Given that connection setup specifications are typically relaxed under failure conditions [3], and that intelligent design and admission algorithms can be used to reduce the amount of time a network is heavily congested, the analysis, though simplified, should be relevant and insightful. Under failure or heavily congested conditions, the delays are likely to increase, but that will occur with any of the schemes considered (perhaps to a somewhat different degree), such that the relative differences in verification delay are likely to still hold to a first approximation.

By focusing on the propagation times and the reconfiguration times, the analysis is purposely looking at the delays inherent to the various schemes. These are the delays that cannot be architected out of the system. Propagation time is a function of the network topology (i.e., the link distances) and the speed of light in fiber. The reconfiguration time is likely limited by physical phenomena as well (e.g., transients). Clearly there could be additional delays depending on the protocols used, the amount of processing power or memory deployed, etc. However, these delays are more controllable.

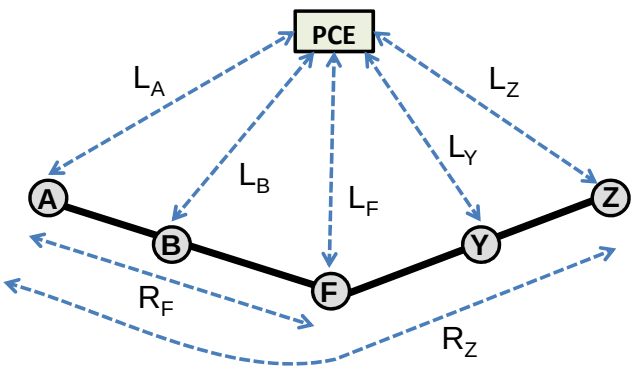

Fig. 1. The new connection path is A-B-F-Y-Z. Node F is assumed to be the furthest from the PCE and is the last to configure its switch.

A. Start Time Based On Timing Mechanism

We first consider the scenario where the transmission start-time at the source is strictly determined based on a timing mechanism. The timing must ensure that when the initial transmission has reached a node along the path that the switch at that node has been configured. In an all-optical path, there is no means of storing the data at a node to allow time for the switch configuration to be completed. (Note that optical burst switching operates with this type of timing mechanism.)

Let Δ represent the time that the source waits after configuring its own switch until it begins transmission. At any node i, the following timing constraint is imposed, which can be seen from the diagram in Fig. 1:

$$L_A + L_A + S + \Delta + R_i \geq L_A + L_i + S \tag{1}$$

Typically, $L_A + R_i \geq L_i$. This holds, for example, if shortest path routing is used, which is common in carrier networks. In this case, Δ is 0; i.e., in the ideal case, the source does not need to wait to begin transmission. Assuming Time 0 is when the source sends the connection request to the PCE, then transmission can commence at time:

$$2L_A + S \tag{2}$$

In an actual implementation, a small delay would be imposed to account for non-ideal conditions (e.g., a longer L_i due to congestion) or differences between the control-plane and data-plane delays (note that it is R_i of the data plane that is relevant in Eqn. 1).

B. Start Time Based On Positive Acknowledgment

Next, we consider the scenario where the source must receive a positive acknowledgment (ACK) of the proper setup from each node in the new path prior to commencing transmission. If a positive ACK is not received after an appropriate delay, or if a negative acknowledgment is received due to, for example, equipment failure, the connection setup is considered failed. We consider two ACK mechanisms below.

1) Centralized Acknowledgment Scheme: In the first mechanism, each node in the path sends an ACK to the PCE upon successful configuration. After receiving an ACK from all nodes, the PCE sends a message to the source indicating it can commence transmission. Under normal conditions, the ACK from Node F is the last to arrive at the PCE. Thus, the source can commence transmission at time:

$$L_A + L_F + S + L_F + L_A = 2L_A + 2L_F + S \tag{3}$$

Compared to Eqn. 2, verification imposes an added delay of:

$$2L_F \tag{4}$$

TABLE I
ASSUMPTIONS FOR PROPAGATION AND SWITCH TIMES (MS)

		S	L_A	L_F	L_Z	R_Z
Back-bone	**Set 1**	15	5	20	17	21
	Set 2	15	10	11	10	8
Metro	**Set 3**	15	1.5	4.2	4	4.5
	Set 4	15	1.6	1.7	1.5	2

TABLE II
CENTRALIZED CONNECTION SETUP: TRANSMISSION START TIME (MS)

		No Verification	Centralized ACK	Distributed ACK
Back-bone	**Set 1**	25	65	58
	Set 2	35	57	43
Metro	**Set 3**	18	26	25
	Set 4	18	22	20

To get an idea of the significance of this delay, we consider the sets of values shown in Table I. The first two rows represent a backbone network scenario, with propagation delays in the 5 ms to 21 ms range. The lower two rows represent a metro network scenario, with propagation delays in the 1.5 ms to 4.5 ms range. The numbers are arbitrary, but were chosen to correspond with realistic topologies, with different extremes of L_F relative to L_A (i.e., the relative distances of the furthest node and the source node in relation to the PCE). If we assume that these delays incorporate 1 ms for queuing/processing, then they correspond to distances of 800 km to 4,000 km in the backbone, and 100 km to 700 km in the metro. The start times under no verification and under verification with a centralized ACK are shown in columns 1 and 2 of Table II, respectively.

Clearly, the percentage increase in delay imposed by the centralized ACK scheme depends on the underlying times and the magnitudes of the propagation times relative to the configuration time. For the values shown, the percentage increase in delay resulting from verification with a centralized ACK mechanism ranges from 60% to 160% in a backbone network and 20% to 50% in a metro network. In a metro network, the configuration delays dominate the propagation delays, such that requiring an ACK is less onerous.

2) Distributed Acknowledgment Scheme: In an alternative ACK-based mechanism, each node along the new path sends an ACK directly to the source. This represents a hybrid scheme, where the setup process is centralized but the verification process is distributed. If we assume that shortest-path routing is used (again, this is typical in carrier networks), then it is sufficient to consider when the ACK from Node Z arrives at the source. This holds because $L_Z + (R_Z - R_i) \geq L_i$ for all intermediate nodes i. The ACK from Node Z arrives at time:

$$L_A + L_Z + S + R_Z \tag{5}$$

Compared to Eqn. 2, distributed verification imposes an additional delay of:

$$L_Z + R_Z - L_A \tag{6}$$

Using the same sets of values shown in Table I, the final column of Table II shows the start time for the distributed ACK scenario. The percentage increase in start time as compared to the unverified scenario ranges from 25% to 130% for the backbone case, and from 10% to 40% for the metro case. Typically, the distributed ACK scheme adds less latency than that added by the centralized ACK scheme (due to $L_Z \leq L_F$ and $R_Z \leq L_A + L_Z$).

III. VERIFICATION IN A DISTRIBUTED ARCHITECTURE

In this section, we consider a distributed architecture for connection setup. The analysis is applicable to distributed setup schemes such as Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) [14] or 3-way Handshake (3WHS) [15]. 3WHS typically yields better performance with respect to blocking as compared to RSVP-TE [16], but is somewhat more complex. The specific scheme that we describe is based on 3WHS to demonstrate how verification can be implemented even with the more complex scheme.

In Pass 1 of the setup protocol, probes are sent along pre-calculated paths to determine the available resources. We assume that the end-to-end latencies of the various probed paths are roughly the same. Carriers prefer not to consider excessively long paths, so this assumption is reasonable. (Note that the probed paths do not have to be mutually link-disjoint. Disjointness with respect to the bottleneck links in a network is sufficient to achieve good load balancing [1].) Additionally, in 3WHS, once some number of successful probes have been received, the destination can commence its processing without waiting for any other probes to arrive.

After the destination evaluates the probes, it selects a path and the necessary resources. It then sends a reservation message back to the source along the selected path; this is Pass 2. This message indicates to the intermediate nodes the resources that are to be reserved for this new connection and the necessary equipment configurations that are required. We assume that pipelining is employed at each intermediate node; i.e., a node forwards the reservation message without waiting for its own equipment to be reconfigured.

In 3WHS, Pass 2 may reserve extra resources to minimize the impact of blocking; i.e., if the desired resource reservation is no longer viable at some node or link along the path due to a concurrent connection request grabbing the resource first, the other resource reservation may still be viable. In essence, Pass 2 attempts to establish two end-to-end connections, using different resources, along the path between the source and destination. Assume that a multicast-drop capable ROADM is used at Node Z; e.g., a ROADM based on the broadcast-and-select architecture. Then the client at Node Z can be cross-connected into both of the connections at the start of Pass 2. The fact that this cross-connect can be triggered at the start of Pass 2 is important for the timing of the verification message. (RSVP-TE creates just one connection; thus, a multicast-capable ROADM is not required to allow the cross-connection to the client to commence at the start of Pass 2.)

The reservation message is eventually received at the source. Assuming that all of the resources corresponding to at least one

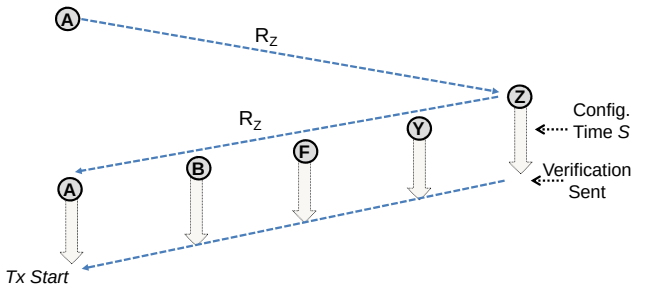

Fig. 2. Node A can start transmission at time $R_Z + R_Z + S$, regardless of whether verification is required.

TABLE III
DISTRIBUTED CONNECTION SETUP: TRANSMISSION START TIME (MS)

		No Verification
Back-bone	**Set 1**	57
	Set 2	31
Metro	**Set 3**	24
	Set 4	19

of the reservation sequences were successfully reserved, then the source selects the preferred resource sequence and begins to configure its own equipment. It also sends a message back along the path to the destination to release any of the resources that were unnecessarily reserved. This release process, Pass 3, does not affect the transmission start time.

The timing diagram for the scheme is shown in Fig. 2. Given the assumption that the configuration time, S, is the same at all nodes, then the source can begin transmission as soon as it completes its own configuration. If we assume Time 0 is when the source first sends out the probes, then transmission starts at time:

$$R_Z + R_Z + S \tag{7}$$

Using the values from Table I, Table III indicates the start time with no verification for the distributed setup architecture, for the backbone and metro network scenarios. Comparing the values of Table III (Eqn. 7) with the first column of Table II (Eqn. 2), the start-up latency with no verification could be greater or smaller than that in the centralized setup architecture depending on the length of the new path as compared to the distance of the PCE from the source.

If verification is required, then the destination node sends an ACK as soon as it is properly configured. By the time that ACK reaches the next upstream node, the equipment at that node should be configured as well (assuming S is the same for all nodes). Ultimately, the ACK is received at the source at time $R_Z + R_Z + S$, which corresponds to when the source completes its own configuration. Thus, in this ideal case, there is no extra latency imposed by requiring positive acknowledgment of the connection path prior to starting transmission. Furthermore, the start time with verified distributed setup is no greater than (and typically less than) the start time of either of the verified centralized setup schemes discussed earlier.

IV. CONCLUSION

We have investigated how requiring a positive ACK of the proper connection configuration affects the setup latency. We showed that the impact is much greater in a backbone network than in a metro network due to the much longer propagation delays in a backbone. We showed that the extra latency imposed by requiring verification is typically smaller in a distributed connection-setup scheme as opposed to a centralized setup scheme, and in fact can be close to 0. Thus, for critical dynamic applications with stringent setup time requirements, distributed operation can be advantageous, especially in a backbone network.

REFERENCES

[1] J. M. Simmons, *Optical Network Design and Planning*, Second Edition, New York, NY: Springer, 2014.
[2] A. A. M. Saleh, "Dynamic multi-terabit core optical networks: architecture, protocols, control and management (CORONET)," Defense Advanced Research Projects Agency (DARPA) Strategic Technology Office, BAA 06-29 PIP, Aug. 2006. Available online at: http://monarchna.com/CORONET-InfoPamphlet-DARPA.pdf.
[3] A. L. Chiu, et al., "Architectures and protocols for capacity efficient, highly dynamic and highly resilient core networks," *J. Opt. Commun. and Netw.*, vol. 4, no. 1, pp. 1–14, Jan. 2012.
[4] A. Von Lehmen, R. Doverspike, G. Clapp, D. M. Freimuth, J. Gannett, A. Kolarov, H. Kobrinski, C. Makaya, E. Mavrogiorgis, J. Pastor, M. Rauch, K. K. Ramakrishnan, R. Skoog, B. Wilson, and S. L. Woodward, "CORONET: Testbeds, demonstration, and lessons learned," *J. Opt. Commun. and Netw.*, vol. 7, no. 3, pp. A447–A458, Mar. 2015.
[5] O. Gerstel, I. Leung, G. Nicholl, H. Sohel, W. Wakim, and K. Wollenweber, "Near-hitless protection in IPoDWDM networks," *Optical Fiber Communication/National Fiber Optic Engineers Conference (OFC/NFOEC'08)*, San Diego, CA, Feb. 24-28, 2008, Paper NWD4.
[6] M. Freiberger and M. T. Watts, "Low latency networks: future service level use cases and requirements," *Optical Fiber Communication Conference (OFC'18)*, San Diego, CA, Mar. 11-15, 2018, Paper Tu2K.4.
[7] A. A. M. Saleh, "Technologies, architecture and services for the next-generation core optical networks," *Optical Fiber Communication/National Fiber Optic Engineers Conference*, Workshop on the Future of Optical Networking, Anaheim, CA, Mar. 25-29, 2007.
[8] C. Westphal, "Challenges in networking to support augmented reality and virtual reality," *IETF 98*, March 26-31, 2017.
[9] K. Shiomoto and A. Farrel, "Advice on When it is Safe to Start Sending Data on Label Switched Paths Established using RSVP-TE," Internet Engineering Task Force, Request for Comments (RFC) 6383, Sep. 2011.
[10] W. Sun, G. Zhang, J. Gao, G. Xie, and R. Papneja, "Label Switched Path (LSP) Data Path Delay Metrics in Generalized MPLS and MPLS Traffic Engineering (MPLS-TE) Networks," Internet Engineering Task Force, Request for Comments (RFC) 6777, Nov. 2012.
[11] Cisco, "Encrypted traffic analytics," Jan. 2018. Avail. at: www.cisco.com/c/dam/en/us/solutions/collateral/enterprise-networks/enterprise-network-security/nb-09-encrytd-traf-anlytcs-wp-cte-en.pdf.
[12] A. Farrel, J.-P. Vasseur, and J. Ash, "A Path Computation Element (PCE)-based Architecture," Internet Engineering Task Force, Request for Comments (RFC) 4655, Aug. 2006.
[13] Q. Zhao, Z. Li, D. Dhody, S. Karunanithi, A. Farrel and C. Zhou, "PCEP Procedures and Protocol Extensions for Using PCE as a Central Controller (PCECC) of LSPs," IETF Draft, June 18, 2018.
[14] G. Swallow, J. Drake, H. Ishimatsu, and Y. Rekhter, "Generalized Multiprotocol Label Switching (GMPLS) User-Network Interface (UNI): Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Support for the Overlay Model," Internet Engineering Task Force, Request for Comments (RFC) 4208, Oct. 2005.
[15] R. A. Skoog and A. L. Neidhardt, "A fast, robust signaling protocol for enabling highly dynamic optical networks," *Optical Fiber Communication/National Fiber Optic Engineers Conference (OFC/NFOEC'09)*, San Diego, CA, Mar. 22-26, 2009, Paper NTuB5.
[16] R. Skoog, J. Gannett, K. Kim, H. Kobrinski, M. Rauch, A. Von Lehmen, and B. Wilson, "Analysis and implementation of a 3-way handshake signaling protocol for highly dynamic transport networks," *Optical Fiber Communication Conference (OFC '14)*, March 9-13, 2014, San Francisco, CA, Paper W1K.1.