Concurrent Processing of Multiple LSP Request Bundles on a PCE in a WDM Network

J. Ahmed, P. Monti, L. Wosinska
The Royal Institute of Technology (KTH), Stockholm, Sweden
{jawwad, pmonti, wosinska}@kth.se

Abstract: Concurrent RWA algorithm for differentiated services to process multiple LSP bundles at PCE is proposed. Significant blocking probability reduction has been observed at the expense of slightly increased LSP setup-time compared to a sequential approach.

OCIS codes: (060.0060) Fiber optics and optical communications; (060.4250) Networks

1. Introduction

Path computation in optical networks can be quite complex and may be subject to multiple constraints such as wavelength continuity, physical impairments, and QoS requirements (e.g., delay, bandwidth, and load balancing). Typically, in a GMPLS-based network, path computation is performed at ingress nodes in a distributed manner to enhance network scalability. Recently an alternative approach has emerged, where a Path Computation Element (PCE) [1] serves as a central entity specialized in solving complex, multi-constrained Label Switched Path (LSP) computation requests. This approach is particularly useful when some nodes in the network do not have the required processing power for distributed path computation operations. The PCE concept was validated in an experimental network [2,3], where the performance of the path computation procedure was evaluated in both single and multi-area networks. Performance was then compared with a conventional, GMPLS-based, distributed path computation approach. Results demonstrated both efficient resource utilization and good network scalability of the PCE-based approach.

In the PCE paradigm, communication between a node and a PCE is specified by the Path Computation Element communication Protocol (PCEP) [4]. The PCEP protocol defines communication semantics between the Path Computation Client (PCC) and the PCE (or multiple PCEs), using PCReq and PCRep messages to send LSP requests/responses from a PCC to a PCE and vice-versa. In a single PCReq message several LSP requests can be *bundled* together before being sent to the PCE. In a similar way several path computation responses can be bundled in a single PCRep message in the opposite direction. This bundling feature can be exploited to achieve significant improvements in terms of overall network performance including the concurrent optimization of a large set of LSP requests with the consequent reduction of both the network blocking probability and the overall control plane overhead. However, in this process there is a tradeoff involved in terms of connection setup delay. An initial assessment of this tradeoff is made in [5]. Pros and cons associated with the bundling approach are studied in more detail in [6], where the concurrent optimization of all the LSP requests bundled in a *single* PCReq message (i.e., originating at the same ingress node) is enabled at the PCE.

In this paper the approach proposed in [6] is extended to concurrently process LSP requests belonging to *multiple* PCReq message. The rationale behind the idea is to further reduce the network blocking probability by allowing the concurrent optimization of LSP requests originating at different ingress nodes of the network. For this reason an additional threshold (i.e., PCReq message counter) is defined at the PCE. This threshold, referred to as *BundleThrsh#*, allows for collecting a certain number of PCReq messages (and consequently bundles of LSP requests) before being concurrently processed at the PCE. Furthermore, a differentiated service traffic environment is assumed where a specified percentage of the traffic arriving in the network requires dedicated, shared and no-protection respectively.

Simulation results, with LSP requests requiring dedicated, shared and no (path) protection, show that by choosing an appropriate value of BundleThrsh# it is possible to significantly reduce blocking probability at the expense of a slightly increased LSP setup-time.

2. Employing a PCReq counter at the PCE

In this study some bundled LSP requests are assumed to be both *synchronized* (from a concurrent optimization point of view) and *dependent* (the computation of the protection path is based on route of the primary path), while others

This work was supported by the Network of Excellence “Building the Future Optical Network in Europe” (BONE), funded by the European Commission through the 7th ICT-Framework Programme and the EUREKA/CELTIC project “Management Platform for Next Generation Optical Networks” (MANGO).

978-1-55752-884-1/10/$26.00 ©2010 IEEE

are assumed to be *synchronized* only (when no-protection is required). For this reason the SVEC (Synchronization Vector) object is used in the PCReq messages. A time-threshold based approach is used to bundle LSP requests at each ingress node, where the PCC bundles LSP requests in a single PCReq message after a timer expires. This approach ensures an upper bound on the waiting time for the LSP requests at each ingress node. In addition, at the PCE, a counter (BundleThrsh#) is employed which ensures that a specific number of PCReq messages are collected before the path computation phase starts. When the required number of PCReq messages is collected, each PCReq message is opened, and all the LSP requests are fed to the concurrent optimization algorithm. After the path computation phase is over, LSP responses are bundled into PCRep messages (based on their respective source node) and sent back with the computed path. This in turn has a beneficial effect on the control plane overhead reduction, particularly with high time-threshold values at the ingress node as well as relatively high values of the BundleThrsh#. However, this optimization is more beneficial when network traffic is highly skewed (i.e. large amount of traffic is generated by only a small number of source nodes).

3. Simulation setup and environment

Results are computed using the POSE discrete event-driven simulator [7]. Simulation results are collected using the European Optical Network Triangular Type (EON-TT) topology [8], which comprises 28 nodes and 61 links. Each link in the network is bidirectional with one fiber and 20 wavelengths for each direction. Wavelength conversion is not allowed, i.e., the wavelength-continuity-constraint is enforced. Two scenarios are considered for LSP request processing: (i) *Sequential*, when all the requests in a PCReq message are processed at the PCE in a sequential manner (i.e., one by one) and (ii) *Concurrent*, when these requests are processed in a combinatorial manner (i.e., a concurrent RWA algorithm). For the sequential case routes are computed using the EWLCR (Enhanced Weighted Least Congested Routing) algorithm [9] and wavelengths are assigned on a First-Fit basis. For the concurrent case the RWA heuristic proposed in [6] is utilized at the PCE. Arrival rate of the LSP requests follows a Poisson distribution and the service time is assumed to be exponentially distributed. A Destination Initiated Reservation (DIR) signaling scheme similar to RSVP-TE is employed for resource reservation after the path computation phase. Considered performance parameters are: blocking probability, PCEP overhead and average LSP setup-time. The starting time, for the threshold-based timers, is randomly offset at each node, to avoid synchronization.

For benchmarking purposes, a *Baseline* scenario is also identified. In this scenario LSP request bundling is not used at all (i.e., the time threshold at each ingress node is set to zero). The estimation of the PCEP bandwidth overhead includes: TCP, IP and Ethernet overhead, assuming that the control plane is implemented over Ethernet. LSP setup-time includes path computation, communication, queuing time and time necessary for reserving the computed path through the network (i.e., signaling time). The total number of LSPs to be established is 50,000. The mean LSP service time is fixed at 60 s, while the arrival rate is assumed to be 1/150 arrivals per second per node pair, which gives a total load in the network equal to 300 Erlang. . For the generated LSP requests we consider differentiated reliability requirements. We assume that on average the following protection requirements: 10% of requests require dedicated protection, 40% require shared protection and 50% do not require any protection at all.

4. Simulation results

Note that *BundleThrsh#* is set to 1 for the *Concurrent* case in Fig. 1 and 2. Fig. 1 shows the effect of LSP bundling on the blocking probability as the time-threshold varies from 1 to 80 s. For the *Concurrent* case, difference with the *Baseline* scenario in terms of blocking probability is almost non-existent until a time-threshold value of 40 s. Blocking probability increases rapidly after 40 s because of too many LSP requests need to be setup at approximately the same time. For the *Sequential* case the trend is similar but with a higher slope. Fig. 1 also shows the gain in terms of control overhead reduction. A 30% reduction is possible when the time-threshold is equal to 40 s. LSP setup-time (Fig. 2) increases linearly with the time-threshold value for both the *Concurrent* and the *Sequential* case. The figure also shows that the gap between the two cases tends to widen for higher values of the time-threshold. This is because more and more LSP requests are bundled in each PCReq message (and need to be computed concurrently at the PCE). This in turns causes the running time of the RWA heuristic to increase, with a detrimental effect on the queuing time of LSP requests at the PCE as well. It can be concluded that under the given network scenario, employing a time threshold of 40 s can result in a significant reduction in control overhead without a major increase in the average LSP setup-time and blocking probability. This value of the time threshold (40 s) is then used to investigate the effect of the BundleThrsh# at the PCE. Fig. 3 shows the value of the blocking probability and the average LSP setup-time as a function of BundleThrsh#. Blocking probability drops almost linearly with increasing values of the counter, but the LSP setup-time increases exponentially. So, a good value of the BundleThrsh# is a trade-off between the blocking probability and LSP setup-time. A value of 3 for the BundleThrsh# seems to be a good enough compromise in the current network scenario. A plot for the PCPEP control overhead is not shown because the effect of BundleThrsh# on that parameter is negligible.

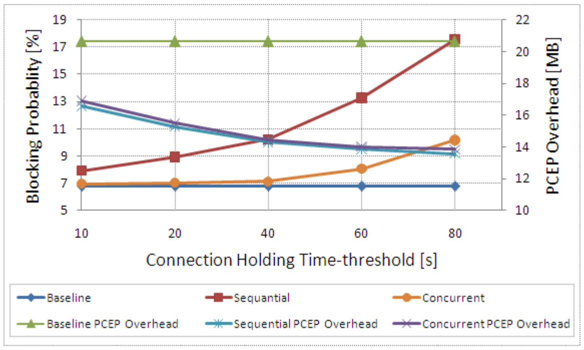

Fig.1. Blocking Probability [%] & PCEP Overhead [KB] vs. connection holding Time-threshold [s].

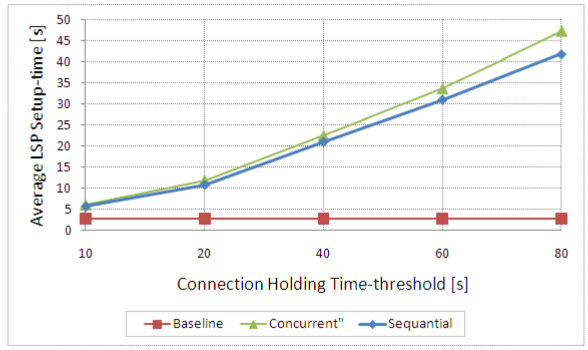

Fig.2. Average LSP Setup-time [s] vs. connection holding Time-threshold [s].

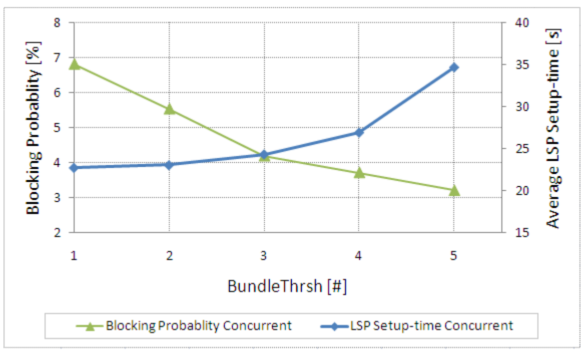

Fig. 3. Blocking Probability [%] & Average LSP Setup-time [s] vs. BundleThrsh [#].

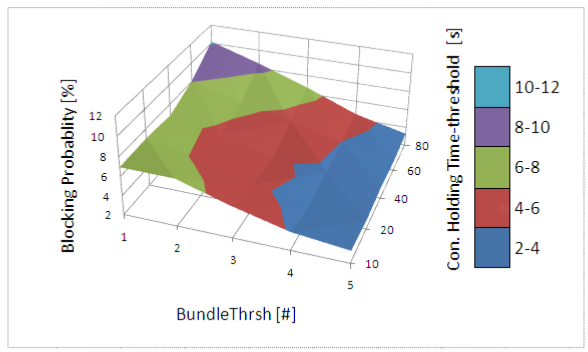

Fig. 4. Blocking Probability [%] vs. Con. Holding Time-threshold [s] vs. BundleThrsh [#].

Fig. 4 characterizes the combined effect on blocking probability of the connection holding-time at the ingress node and the BundleThrsh# at the PCE. It can be observed that blocking probability is strongly correlated to connection holding-time threshold and BundleThrsh# under any specific network scenario.

5. Conclusion

In this study the possibility of concurrent processing of multiple PCReq messages at the PCE was explored in order to reduce blocking probability in a PCE based WDM network by optimizing the utilization of network resources. Results show that, by choosing an appropriate value of the BundleThrsh# a significant reduction in blocking probability can be achieved without a noticeable increase of the LSP setup-time.

References

[1] A. Farrel, J.-P. Vasseur, and J. Ash, “A Path Computation Element (PCE) Based Architecture”, IETF RFC 4655, Aug. 2006.
[2] F. Cugini, et al., “Implementing a Path Computation Element (PCE) to encompass physical impairments in transparent networks”, OFC/NFOEC 2007.
[3] P. Castoldi, et al., “Centralized vs. Distributed Approaches for Encompassing Physical Impairments in Transparent Optical Networks”, ONDM 2007.
[4] J.-P. Vasseur and J. L. Le Roux, “Path Computation Element (PCE) Communication Protocol (PCEP)”, Internet draft, Nov. 2008.
[5] J. Ahmed, et al., “LSP Request Bundling in a PCE-Based WDM Network”, OFC/NFOEC 2009.
[6] J. Ahmed, et al., “Benefits of Connection Request Bundling in a PCE-based WDM Network”, NOC/OC&I 2009.
[7] J. Ahmed and S. Nawaz, “POSE: A New Discrete Event Optical Simulator for the Java Platform,” NCET 2004.
[8] S. de Maesschalck et al., “Pan-European Optical Transport Networks: An Availability-based Comparison,” Photonic Network Communications, 5(3), May 2003.
[9] J. Ahmed, et al., “Dynamic routing in wavelength convertible WDM networks,” INMIC 2004.