Dynamic Prioritization of Path for Quality-of-Service Differentiation in Multi-Priority Traffic

Fatemeh Banaie[1], Seyed-Amin Hosseini-Seno[1], Mohammad Hossein Yaghmaee[2] and Rahmat Budiarto[3]

[1]Computer Network Research Laboratory, Ferdowsi University of Mashhad, IRAN

[2]Department of engineering, Ferdowsi University of Mashhad, IRAN

[3]College of Computer Science and Information Technology, Al Baha University, Kingdom of Saudi Arabia

banaie_f@yahoo.com, hosseini@um.ac.ir, yaghmaee@ieee.org, rahmat@bu.edu.sa

Abstract—**The emergence of value added services relying on a higher interactivity has altered the requirements of current transport network. Diverse traffic classes are processed by a large-scale optical network, imposing a more efficient utilization of their network infrastructure resources. Such services generally cross multiple domains, but inter-domain path computation algorithms still have some limitations. This paper describes a priority based path computation algorithm to meet all QoS requirements with the available capacity. The proposed algorithm increases the rate of successful replies while minimizing the blockage in network. The dynamic traffic is classified into high and low priority, so it improves emergency response in network.**

Keywords-Path computation; inter-domain; traffic; Priority

I. INTRODUCTION

Reliable delivery is an obligatory requirement of current transport networks. Advances in networking technology and increasing demand for QoS guaranteed applications, led to the introduction of traffic engineering techniques to handle network resources in the most flexible way and provide quality of service. One of the main challenges with traffic engineering is to compute a reliable end-to-end path from source to destination. In this context, current transport technologies like Multi-Protocols Label Switching (MPLS) and Generalized MPLS MPLS) are the best protocols for supporting the traffic engineering.

With expanding the scenario from single-domain to multi-domain networks, some challenges arise, such as restricted topology visibility due to scalability issues. In multi-domain networks, routing domains managed by different network providers, have their own routing policies and information [1]. As a result, computing optimal routes presents a huge problem, due to the need for preserving information confidentiality across domains. In particular, inter-domain traffic engineering (TE) techniques are not currently implemented by network providers and provisioning of QoS for applications across multiple domains is performed manually [2]. So, they require efficient mechanisms to perform end-to-end path computation between source and destination nodes belonging to different administrative domains.

In this context, the Internet Engineering Task Force (IETF) has proposed a set of techniques defined based on the Path Computation Element (PCE) Architecture [3]. In such techniques, routing decisions in each domain is delivered to dedicated network entities (e.g., the PCEs).

The main objective of this paper is to propose a dynamic priority assignment mechanism for path computation process across multiple domains. This solution investigates the integration of request priorities into the path computation method and analysis the network overall utilization and successful reply rates of requests in different priority categories. The proposed distributed mechanism not only finds the inter-domain paths, but also ensures the deployment of Layered Service Provider (LSP) without blocking.

The rest of this paper is organized as follows. Section II provides related work on the topic. In section III, the proposed mechanism is described in detail. Section IV presents and discusses the simulation results. Finally, Section V gives a conlcusion and further research directions.

II. RELATED WORK

Considerable research activity has been focused on the inter-domain path computation schemes in the recent years. We can categorize these researches into two classes from the perspective of routing architecture and protocol [4]. The first class focuses on extending the BGP protocol to carry QoS information. The literature has proposed some extensions for performing inter-domain path computation using Border Gateway Protocol (BGP) [5-6]. But the current release of BGP is inadequate for most inter-domain applications, due to the lack of QoS routing capabilities and scalability issues. The second class attempts to propose new architecture for inter-domain path computation. One of the most considered architectures is PCE architecture that has been introduced to calculate end-to-end routes with QoS constraints in single and multi-domain networks.

PCE is an entity which can be implemented at a network element or can be as an independent entity. PCE performs constraint-based path computation using its traffic engineering data base (TED) in a single domain, but it has limited routing information from other domains (Fig. 1). It also communicates with other domains using Path Computation Element Communication Protocol (PCEP) [7].
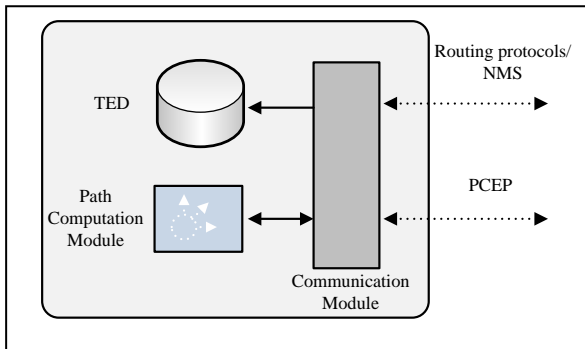
Figure 1. Functional modules of a Path Computation Element [2].

The work in [8] classifies inter-domain path computation based on PCE into two categories: model-based approaches and ad-hoc approaches.

Model-based approaches assume that the sequence of domains to be traversed is determined before the path computation process. Looking at standardization, the reference model-based approaches are per-domain path computation (PD) [8] and Backward Recursive PCE-based computation (BRPC) [9].

In per-domain path computation method, every intermediate domain is assumed to have a PCE and computes individual path segments without sharing of any path information from other domains. Then, path segments computed for every domain are joined to obtain the complete path. This procedure may lead to sub-optimal path computation, due to miss-coordination between domain PCEs.

In BRPC method, a constrained shortest path is computed, in a reverse fashion, from the destination domain towards the source domain. Upon receiving a request message, PCE in the destination domain creates a virtual shortest path tree (VSPT) and sends it back to PCE in previous domain. Then each intermediate PCE adds its local path information to the received VSPT and sends it back to the previous domain PCE, until it reaches to the requesting PCC. PCC selects the optimal end-to-end path from the tree.

Model-based approaches assume that the domains sequence to be traversed is known in advance. Since, in complex multi-domain networks, the selection of the domain sequence considerably affect the overall network performance, we are trying to defining a procedure combining domain sequence computation and path computation.

In the case that domain sequences is not known, [10] introduced Path Computation Flooding (PCF). In PCF, the source domain PCE sends the path computation request to all neighboring domains until it reaches to destination PCE. The destination PCE computes its local VSPT and sends it back to all adjacent domains. In the same way, all intermediate PCEs add their local path information to VSPT and send it back to neighboring PCEs. This flooding approach continues until source PCE receives all VSPTs from all neighboring domains. Then it can select the optimal path among these possible end-to-end routes. Clearly, this procedure has considerable scalability problem and network overhead that lead to discard this approach for large multi-domain networks.

The work presented by [11-13], extended PCE architecture to combine domain sequence computation and path computation. These methods use the hierarchical relationship between domains to select the optimum sequence of domains. In the hierarchical PCE, a parent PCE is responsible for inter-domain path computation and child domains perform intra-domain path computation. However, hierarchical PCE model is appropriate for environments with small groups of domains, and it is not applicable to large groups of domains such as Internet [14].

III. PROPOSED MECHANISM

Aiming to address aforementioned path computation limitations, we propose a priority-based path selection for path computation process across multiple domains. But, most of the inter-domain path computation procedures pose a significant PCE response time that could result in blockage during deployment time. So, we propose to use a path computation procedure based on the pre-reservation of the resources dedicated to the path. This procedure uses PCE architecture for inter-domain path computation and has a considerable effect in reducing the blockage [15].

A. PCE-based Multi-domain Network Model and Path computation

We model the network as a graph $G(V, E)$, where V and E are sets of nodes and links, respectively. This global graph consists of D sub-graphs, $G^i = \{G^1, \dots, G^D\}$, where each sub-graph presents one domain and D is the total number of domains.

In this model, we use *Request* and *Reply* messages to send a path request and reply for it. *Request-confirm* is also used to confirm recipient of messages. At first, PCC (path computation client) sends a *Request* message to the source PCE. PCE forwards this message for all of its neighboring PCEs, until it reaches the destination PCE. When a request passes through intermediate PCEs, PCEs check its *ReqID* with their own entry tables. If the *ReqID* does not exist in the table, they record *ReqID* and cost of that request in their tables. When *ReqID* is already registered, PCE compares the new received cost with the registered cost in its entry table. If it is greater than the previous cost, the received message will be deleted; otherwise, the cost of the new message will be replaced with the old one in its table. Upon receiving the request message by the destination PCE, the required resources are compared with the available resources. If the available resources are not sufficient, a *PErr* Message will be returned. Otherwise, it computes all possible paths in its domain and adds it to the *Reply* message. Then, it pre-reserves required resources and sends the *Reply* message to the previous domain PCE. The *Rely* message is returned upstream to the source PCE. Pre-reserved

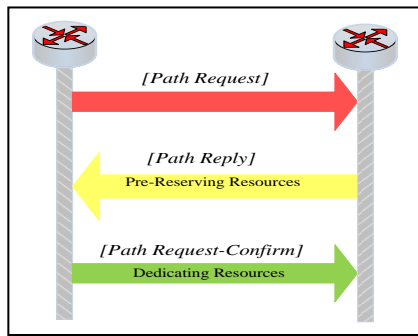resources will be dedicated when each PCE receives the *Request-confirm.* (Fig. 2).

Figure 2. Resource Reservation Mechanism.

In this method, each PCE requires to keep *ReqID* and cost of the requests. So, it can be argued that this method consumes more memory for recording the request's information. However, as we prune routes in network and this reduces the amount of traffic in the whole path. As a result we record the information in limited number of nodes and do not need to record them in all of PCEs. On the other hand, we need to keep connection information and resource reservation while setting up a connection that requires local memory. In case of flooding the messages, we have higher connections and need much more memory for keeping these paths setup information.

B. *Priority-based Multi-Domain Path Computation Mechanism*

The main objective of the proposed method is to disseminate requests in a multi-domain network in order to meet all QoS requirements with the available capacity. The proposed method uses pre-reservation technique, as introduced by [15], in path computation process and defined under the umbrella of PCE architecture. As mentioned previously, in inter-domain path computation scenarios, there is a considerable time interval from sending a request until receiving a reply for it. So, by the time the signaling for the LSP deployment reaches the corresponding domains, resources that were previously available during the path computation process may not be available anymore. This procedure increases the probability of deployment failure in complex inter-domain path computations. By pre-reserving resources, we ensure that required resources from the source PCE to the destination PCE are reserved before the actual deployment of LSP.

When calculating the route, communication resources may become congested and cause further path requested to be rejected. In this condition, people with emergency activities may not coordinate their efforts. A high-urgency request should be able to interrupt lower-priority attempts and use their resources. In order to have appropriate resource allocation, it is essential to prioritize access to resources during path computation.

The requirements of individual path request are included into the *Request* message. Dynamic prioritize are calculated at each PCE in the routing path according to the requirements, and current PCE resources. The importance of the path requests in a PCE_j can be ordered as two different priority types: *high priority* and *low priority*. Upon receiving a request with higher priority, it can undo pre-reservations of a lower priority in cases where the available resources are not sufficient. Before allocating, resources are pre-emptible. We also define a threshold for undoing pre-reserved resources to prevent starvation, when the requests with higher priority arrive more frequently. We consider a constant value for threshold, but it can be set by network conditions and rates of service to higher and lower priority requests. In the following we describe the steps of the resource reservation in two different priorities.

Step1: At first, the algorithm compares the required resources with available resources. If there are enough resources to service the request, it will be pre-reserves resources for that request and sends reply message to upstream PCE. Otherwise, it will check the request priority.

$$
\begin{aligned}
&\textit{If available resources are sufficient}\\
&\qquad \textit{pre} - \textit{reserve resorces}\\
&\qquad \textit{Send [Path Reply] to upstream PCE}\\
&\textit{else}\\
&\qquad \textit{check request priority}\\
&\textit{end if}
\end{aligned}
$$

Step2: According to request priority and threshold, the algorithm decides to interrupt the pre-reservation process.

$$
\begin{aligned}
&\textit{If priority is High}\\
&\quad \textit{If} \quad \textit{preemptable resources} < \textit{threshold}\\
&\qquad\qquad \textit{Select a low-priority request}\\
&\qquad\qquad \textit{Undo its pre-reservation}\\
&\quad \textit{else}\\
&\qquad \textit{Send ErrMessage to upstream PCE}\\
&\quad \textit{end if}
\end{aligned}
$$

IV.   SIMULATION RESULTS

In order to evaluate the performance of the proposed protocol, we implemented the priority based proposed path computation mechanism in Opnet v.14 simulator [16]. For this purpose, a network topology as shown in Fig.3 is simulated.
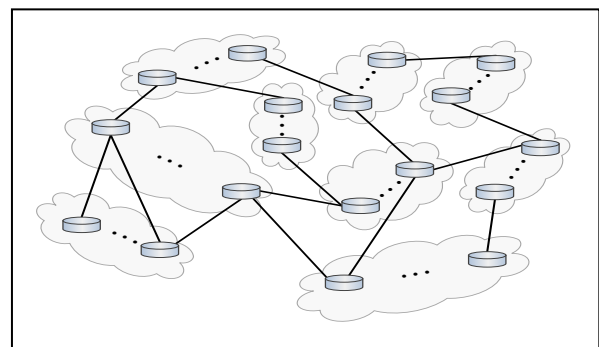

Figure 3. Simulation Topology.

We use *Reply success ratio* and *Network utilization* parameters in two different priorities to evaluate proposed mechanism:

- *Reply success ratio:* The rate of successful replies to the maximum number of requests.
- *Network utilization:* The rate of successful LSP

A. Implementing priority-based path computation mechanism

With the implementation of this procedure, there is a reduction in the number of routes due to the pruning of the non-feasible and non-promising routes. In flooding method, we have to pre-reserve resources in all of these routes. Pruning non-feasible paths decreases the number of reserved resources in network. Thus, other requests can use these resources and we have more resource availability in network.

As seen in Fig.4 and Fig. 5, we have a relative improvement in successful reply ratio for higher-priority requests. That is because of undoing pre-reservations of a lower priority requests by higher ones in cases where the available resources are not sufficient
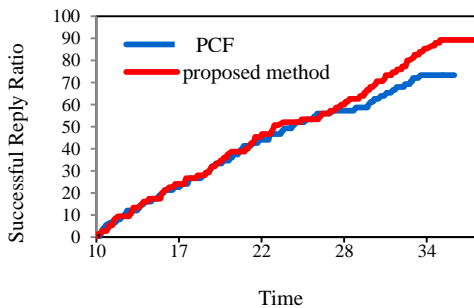

Figure4. Successful Reply ratio for high-priority requests.
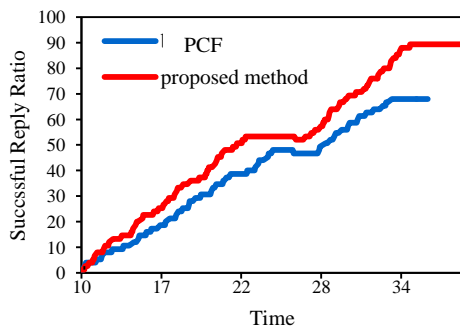

Figure 5. Successful Reply ratio for low-priority requests.

As Fig. 6 and Fig. 7 show, we have an improvement in the network utilization. Because in this case, we have more successful deployment of LSPs, and according to definition of the network utilization, this may increase the network utilization.
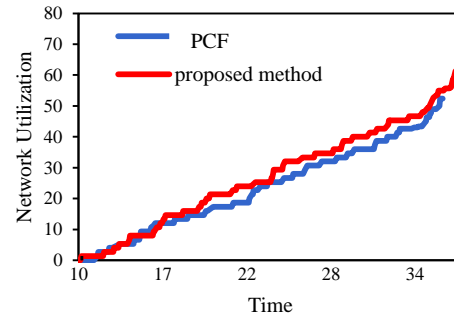

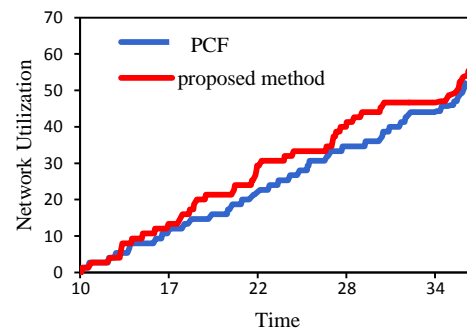Figure 6. Network Utilization for high-priority requests.

Figure 7. Network Utilization for low-priority requests.

V.   CONCLUSION

A variety of traffic classes with different QoS requirements is carried by a large-scale multi domain network. A study on various related works suggest some path computation schemes, but all of them have some limitations (e.g., scalability, confidentiality, domain sequences). In this paper, a distributed priority based path selection algorithm has been developed, which distributes network bandwidth among the data items according to their relative QoS demands. It improves emergency response in network by prioritizing requests.

On the basis of the simulation results, it can be observed that the distributed priority based path selection algorithm has reduced blocking probability by means of increased successful replies and network utilization, especially in higher priority requests.

To continue the work presented in this paper, a mechanism may be introduced to investigate the impact of dynamic changes in preempting threshold value. In other words, instead of setting a fixed value for threshold, we can dynamically change it according to the network conditions such as the service rate of high and low priority requests and etc.

REFERENCES

[1] J.L. Roux, J.P. Vasseur, J. Boyle, "Requirements for inter-area MPLS traffic engineering," Internet Engineering Task Force (IETF) RFC 4105, 2005.

[2]   F. Paolucci, F. Cugini, A. Giregetti, N. Sambo, P. Castoldi, " A survey on the path computation element (PCE) architecture," IEEE Communications Surveys & Tutorial, in press, 2013.

[3] A. Farrel, J. P. Vasseur, and J. Ash, "A path computation element (PCE) based architecture," Internet Engineering Task Force (IETF), RFC 4655, Aug 2006.

[4] R. Fu, "Next Generation Network Routing and Control Plane," Technical University of Denmark, Februuary 2011.

[5] O. Bonaventure, S.D. Cnodder, B.Quoitin, R.White, Controlling the redistribution of BGP routes, Internet Engineering Task Force(IETF), 2003.

[6] O. Bonaventure, B. Quoitin, Common Utilization of the BGP Community Attribute, Internet Engineering Task Force (IETF), 2003.

[7] J. Leroux, "Path Computation Element Communication Protocol (PCECP) Specific Requirements for Inter-Area MPLS and GMPLS Traffic Engineering," IETF, RFC4927, 2007.

[8] J.P. Vasseur, A. Ayyangar, R. Zhang, "A per-domain path computation method for establishing inter-domain traffic engineering (TE) label switched paths (LSPs)," IETF, RFC 5152, February 2008.

[9] J.P. Vasseur, R. Zhan, N. Bitar, J. Le Roux, "A backward recursive PCE based computation (BRPC) procedure to compute shortest constrained inter-domain traffic engineering label switched paths," IETF, RFC 5441, April 2009.

[10] D. King, A. Farrel, "The application of the PCE architecture to the determination of a sequence of domains in MPLS & GMPLS, " Internet Engineering Task Force (IETF) , July 2010.

[11] D. Siracusa, S. Grita, G. Maier, A. Pattavina, F. Paolucci. F. Cugini, P. Castoldi," Domain sequence protocol (DSP) for PCE-based multi-domain traffic engineering," Journal of Optical Communication and Networking, November 2012, Vol. 4, No. 11.

[12] Sh. Shang, N. Hua, L. Wang, R. Lu, X.. Zheng, H. Zhang, "A hierarchical path computation element (PCE)-based k-random-paths routing algorithm in multi-domain WDM networks," Optical Switching and Networking, 2011, 235-241.

[13] L. Buzzi, M. Bardellini, D. Siracusa, G. Maier, F. Paolucci, F. Cugini, L. Valcarenghi, and P. Castoldi, "Hierarchical border gateway protocol (HBGP) for PCE-based multi-domain traffic engineering," in Proc. of ICC Conf., May 2010.

[14] D. King, A. Farrel, "The application of the path computation element architecture to the determination of a sequence of domains in MPLS and GMPLS," Internet Engineering Task Force (IETF), RFC 6805, November 2012.

[15] M. Shirazipour , S. Pierre, " End-to-end path computation schemes for traffic engineering in next generation multi-domain networks," Ph.D thesis, Montreal University of Canada, Juin 2010.

[16] OPNET Technologies, Inc., Opnet Modeler, http://www.opnet.com.