Multi-Constrained Path Computation for Inter-Domain QoS-capable Services[1]

Nabil Bachir Djarallah and Hélia Pouyllau

Alcatel-Lucent Bell Labs France,
Centre de Villarceaux,
91620 Nozay, France.
E-mails: nabil.djarallah@gmail.com,
helia.pouyllau@alcatel-lucent.com

Samer Lahoud and Bernard Cousin

IRISA, University of Rennes 1,
Campus de Beaulieu,
35042 Rennes, France.
E-mails: {samer.lahoud, bernard.cousin}@irisa.fr

Abstract: Computing inter-domain MultiProtocol Label Switching Traffic Engineering Label Switched Path (MPLS-TE LSP) using the Path Computation Element (PCE) through a pre-determined sequence of domains is quite straight. Each PCE, using the Backward Recursive PCE-based Computation (BRPC), knows who is the next to be contacted in order to continue the computation. The optimality of the inter-domain MPLS-TE LSP path depends strongly on the choice of the pre-determined sequence of domains on which the calculation works. In this paper we propose a novel procedure allowing a forward discovery of multiple inter-domain sequences and the computation of constrained inter-domain paths for MPLS-TE LSPs over these domains sequences. Other issues around the inter-domain path computation, such as route discovery and inter-domain loop avoidance, are investigated. Experimental evaluation shows that our solution is effective in terms of protocol and algorithmic efficiency and provides satisfiable performance with high success rate, reasonable message overhead and runtime.

Keywords: Constrained path, MCOP, QoS, PCE, BRPC, MPLS-TE LSP, Inter-domain loop avoidance, Inter-domain route.

Reference

Biographical notes: Nabil Bachir Djarallah received the Ph.D degree in computer science from the University of Rennes, in France. He received also the diploma of computer science engineer from Batna University in Algeria and an M.S. degree in computer science from the University of Rennes, in France. Within the framework of his M.S. degree and Ph.D, he spent almost four years at Alcatel-Lucent Bell Labs France as a research engineer. His research interests include optimization, algorithms for constrained path computation, architectures and inter-carrier service delivery with assured QoS.

Hélia Pouyllau is a Research Scientist at Bell Labs France since January 2008. She graduated from the University of Paris Dauphine in 2004, and then obtained her Phd in computer science in 2007 from the INRIA. Her research interests focus on optimization, learning, distributed algorithms, heuristics, QoS negotiation and provisioning, economics in networks, and inter-domain architectures.

Bernard Cousin is a Professor of Computer Science at the University of Rennes 1. Bernard Cousin received in 1987 his Ph.D degree in computer science from the University of Paris 6. He is, currently, member of IRISA (a CNRS-University joint research laboratory located at Rennes). He is at the head of a research group on networking. He is the co-author of a network technology book: "IPV6 Théorie et Pratique" (Fourth edition, O'Reilly) and has co-authored a few IETF drafts, for instance, in the areas of Explicit Multicasting and Secure DNS. He has co-authored more than one hundred of papers published in international journals and conferences including IEEE Journal on Selected Areas in Communications, Computer Networks Journal, ACM NOSSDAV, IEEE Infocom, IEEE Broadnets, and IEEE Globecom. His research interests include all-optical networks, dependable networking, high speed networks, traffic engineering, multicast routing, network QoS management, network security and multimedia distributed applications.

Samer Lahoud received the Ph.D degree in computer science from Telecom Bretagne, Rennes. After his Ph.D. he spent one year with Alcatel-Lucent Bell Labs Europe working as a research engineer. Since 2007, he has been with the University of Rennes I, where he is working as an assistant professor, and with IRISA of Rennes, where he is taking part in the research activities. His main research activity is in network design, combinatorial optimization and engineering algorithms for communication networks. He has been involved in many research programs at the national and the European level.

1 Introduction

In general, intra-domain optimal constrained path computation is less complex than an inter-domain path computation as each node in the network has a global view on the IGP area topology as well as its Traffic Engineering (TE) information. However, optimal inter-domain path computation presents a challenge because of detailed information filtered at the domain boundary for multiple reasons such as scalability and confidentiality of topological and resources information. One solution to establish inter-domain MPLS-TE LSP subject to multiple Quality of Service (QoS) parameters (e.g. bandwidth, delay, jitter, availability, and loss), is to use the Path Computation Element (PCE) framework (Farrel et al. (2006)).

The PCE framework is a promising technology introduced to solve the problem of multi-constrained path computation in multi-layer, multi-area and multi-domain MPLS and GMPLS networks. Thus, it drives the establishment of inter-domain MPLS-TE LSPs. The PCE framework may compute the end-to-end path itself if enough topology and resource information are available. Alternately, it may opt to compute a part of the path and request another PCE, using a PCE Communication Protocol (PCEP) (Vasseur et al. (2009)), to continue the computation. Thereby, the PCE framework achieves path computation over a larger scope than a usual network node. The PCEP protocol carries the path computation request, in a bilateral cascading manner.

The PCE includes also the Backward-Recursive PCE-Based Computation (BRPC) Procedure (Vasseur et al. (2009)) to compute paths for MPLS-TE LSPs in an intra-domain or inter-domain context. Notice that, BRPC allows the computation of a shortest constrained path among **one sequence of domains** toward the destination domain and assumes that this sequence of domains is **pre-determined** with external mechanism (see section 2). When the sequence of domains is pre-defined, and is not the most appropriate one, the major limitation is that the BRPC procedure cannot guarantee that the MPLS-TE LSP inter-domain path is optimal. In fact, this could happen if one inter-domain sequence is explored.

In addition to this limitation, BRPC does not manage business issues. Thus, to investigate the potential business issues that could be associated to the PCE, we presented a bottom-up approach (Djarallah et al. (2009)) which aims to accommodate business objectives and network resource usage within a business-driven PCE. Therefore, to improve the chances of finding such end-to-end constrained paths, multiple sequences of domains should be explored. In this paper, we propose a protocol that solves the above issues. This protocol allows an automatic exploration of different sequences of domains (inter-domain routes). This paper provides also a way to avoid inter-domain loops that could appear during the exploration of the different inter-domain sequences.

But the protocol itself cannot calculate the end-to-end optimal paths. It can only build a Virtual Shortest Path Tree (VSPT) and convey it from one PCE to another. In contrast, the multi-constrained path computation is provided by the algorithmic part of the PCE. Therefore, the problem of computing QoS-constrained paths is known as the Multi-Constrained Path (MCP) problem (Jaffe (1984)). Its extension to an optimization problem, called Multi-Constrained Optimal Path (MCOP) problem has also been extensively studied in the literature. This paper also aims at solving this problem in the inter-domain context where the objective function (e.g. generated profit, path cost, etc.) would be agreed among a set of federated domains, and where multiple inter-domain routes would be explored as explained before. Furthermore, to be compliant with operators requirements on confidentiality, we intend to provide an efficient distributed algorithm that solves accurately the MCOP problem in a multi-domain context. Efficiency is strictly translated into "optimality" to denote paths that achieve efficient utilization of the network infrastructure resources.

The remainder of the paper is structured as follows: in section 2, we review the most related studies. In section 3, we highlight some definitions and we give a formal definition of the inter-domain multi-constrained path computation problem. Then, we present the algorithm that solves this problem. Section 4 details some challenges related to the inter-domain multi-path exploration and also presents a new protocol for multi-constrained paths over several inter-domain routes. In section 5, we evaluate by simulation the performance of our proposal. Finally, our main conclusions are drawn in section 6.

2 Related work

In this section we'll discuss, first, some related work around the construction of the inter-domain sequence (chain of PCEs) used by PCE in order to reach the target domain (PCE). Second, we review related work on multi-constrained path computation algorithms.

One basic solution to obtain reachability information about remote PCEs, could be based on inter-domain (Internet) routing protocol, as is done today with networks' equipments. However, the *de facto* inter-domain routing protocol, Border Gateway Protocol (BGP) (Rekhter et al. (2006)), returns one path per-destination or prefix, which could be used by the BRPC procedure. In this case, the returned constrained path by BRPC may not be optimal or even feasible if the used inter-domain sequence is not well chosen, hence the importance of exploring multiple routes. In addition, in some cases such as inter-domain load balancing and inter-domain shared path protection, multiple sequences of domains should be explored to compute multi-constrained paths.

Vasseur et al. (2009) suppose that, the domain sequence (PCE chain) is pre-determined by undefined means. This gives network administrators the ability to define and choose the appropriate way to construct inter-domain sequences. One solution (Vijayanand et al. (2007)) is to use the BGP for carrying PCE discovery information (Le Roux et al. (2009)). Boucadair et al. (2005) use the inter-domain routing protocol to announce PCE unique identifiers across the Internet in order to enable other PCEs to discover possible paths towards every domain containing a PCE. Another work proposed by Chen (2006), gives to PCEs the possibility to use information contained in the Autonomous Systems path (AS_PATH) attribute to find out PCEs, track their sequence and to know which PCEs are engaged to compute such path.

All these solutions assume that the PCE runs BGP protocol and uses BGP-based routes to reach remote PCEs. However, the choice remains very limited due to the selection process of paths used by BGP (after filtering, only one path is announced per-prefix). Moreover, these announced BGP paths are not QoS-driven

(Sheldon (2001)). This may not be of interest, to network administrators, because the PCE is used basically to find paths that meet certain QoS requirements.

King et al. (2011) introduce a PCE hierarchy scheme to solve both scalability and domain discovery problems, but do not describe neither how commercial constraints are taken into account nor if this mechanism allows to provide a multiplicity of inter-domain sequences or not. The discovery of a single sequence of domains limits potential traffic engineering features (e.g. no quality of service price/efficiency optimization, inter-domain shared route protection, inter-domain load balancing, etc.).

However, for the multi-constrained path computation problems, satisfying the QoS demands have been long studied, particularly in a mono-domain context (Kuipers et al. (2004); Sanguankotchakorn (2010); Van Mieghem et al. (2004); Shuchita et al. (2010)). Ziegelmann (2007) compare different recent methods both theoretically and experimentally. Other works have addressed the same problems within the inter-domain specificities. Sorte et al. (2002) presented a minimum price inter-domain routing algorithm based on min-plus convolutions that selects the cheapest paths among a number of independent domains satisfying the end-to-end QoS constraints. Algorithms based on exchanging link state information are listed by Norden (2005) and Tae-Il et al. (2009). These algorithms have in common that they exchange some QoS information about inter-domain links and on intra-domain links, in a special case. Two major problems are faced here : confidentiality problems between different domains when QoS information are conveyed and the colossal amount of information that could be exchanged between routers. Norden (2005) presented another algorithm based on parallel probe packets sent through the network to collect routing information. Two levels of packet probing are identified; intra-domain probe packets and the inter-domain probe packets. The intra-domain information collected probe packets are aggregate and sent to the downstream domain. In doing so, networks preserve their confidentiality. Other solutions like those proposed by Bertrand et al. (2009), Amari et al. (2010) propose to solve the inter-domain MCP/MCOP problems using an extended SAMCRA's algorithm (Van Mieghem et al. (2001)). These solutions work on one inter-domain route assumed known in advance. In this paper, we are interested on solving the problem through different inter-domain routes instead of only one pre-determined route. Therefore, we propose a distributed algorithm that computes constrained end-to-end paths over multiple inter-domain routes and takes into account the architecture we proposed in (Djarallah et al. (2009)).

3 Definitions, problem, and algorithm

3.1 Definitions

In order to define the computation problem of multi-constrained paths over multiple domain routes, we model the N interconnected domains as a set of valued graphs, $G_0 = (V_0, E_0) \times ... \times G_i = (V_i, E_i) \times ... \times G_N = (V_N, E_N)$, that form one global network G modeled by a valued graph $G = (V, E)$. Each graph G_i includes a vertex set $V_i(G_i)$ and a set of edges $E_i(G_i)$. The graph G includes a vertex set $V(G)$ and a set of edges $E(G)$ where $E \subseteq V \times V$. To simplify notations we use E_i, E, V_i and V instead of $E_i(G_i), E(G), V_i(G_i)$ and $V(G)$. Vertices represent network nodes, while edges represent communication links.

In the following, we only consider connected graphs without self-loops (an edge which starts and ends at the same vertex) and at most one link between a pair of nodes. A specific link in the set E between nodes u and v is denoted by $e = (u, v)$. Each link $e = (u, v) \in E$ is characterized by an m-dimensional weight vector $w_e[j] = (w_{e,0}, ..., w_{e,m})^T$, where $w_{e,j} > 0$, $\forall\ e \in E$, $j \in [0...m]$ and $m \in \mathbb{N}$. The m components of the weight vector model the m QoS metrics associated to each network communication link, such as bandwidth, availability, delay, jitter, etc.

Multi-constrained inter-domain path. Is an end-to-end path, denoted $P_{s,t}$, between the network source node s and the destination node t, that crosses at least two domains and satisfies the constraint vector $C[j]$. The path $P_{s,t}$ is a finite sequence of path segments. A path segment can be a simple link between two adjacent nodes or an aggregate of several links within the same domain. Each path segment is characterized by a weight vector $w_{e,j}$, where $e \in P_{s,t}$. The e2e multi-constraints vector $W_{P,j}$ associated to the inter-domain path $P_{s,t}$ is composed of the different weight vectors $w_{e,i}$, where $e \in P_{s,t}$.

Non-dominated paths. The paths meeting a request $q(s, t, C)$ demanding an inter-domain path that respects constraints $C[j]$ where $j \in [0...m]$ between source node s and target node t in the graph G, are called **feasible paths** and denoted $P^+_{s,t}$. In order to reduce the search space and to keep only a sub-set of feasible paths, Cormen et al. (1991) used a non-dominance rule. According to this rule, a path p_1 can be discarded when there exists a path p_2 such that $W_{p_2,j} \leq W_{p_1,j}$, for all $j \in [0...m]$, except for at least one j for which $W_{p_2,j} < W_{p_1,j}$. The non-dominance rule is applied on all nodes, during the path computation phase, to discard dominated paths. We denote $P^*_{s,t}$ the set of non-dominated paths found on the destination node.

3.2 Problem statement

The multi-constrained inter-domain path computation over multiple domain routes problem can be defined as finding paths that obey to the constraints vector $C[j]$ (where $C[0]$ is the bandwidth constraint and the other weight components are the QoS additive metrics) and respect the non-dominance rule, from the source node s to the destination node t over a set of inter-domain routes S.

When an optimal path is required, an optimization can be performed in order to identify the optimal path $\hat{p}_{s,t}$. The selection of such path is done through an objective function $Z(P^*_{s,t}) = \{z(p) \setminus \forall p \in P^*_{s,t}\}$. This function can take several forms according to the policy adopted by participant domains (e.g. path cost, generated profit, remainder bandwidth, etc.). This is known as the Mult-Constrained Optimal Path problem (MCOP). In our inter-domain context we call this problem, Inter-Domain Multi-Constrained Optimal Path Over Multiple Domain Routes (ID-MCOP-MDR) problem. Consequently, the ID-MCOP-MDR problem can be expressed as follow:

$$\min_{\forall p \in P^*_{s,t}} Z(P^*_{s,t}) \tag{1}$$

subject to,

$$\forall\ p \in P^*_{s,t},\ W_{P,j} = \sum_{\forall e \in p, j=1}^{m} w_{e,j} \leq C_j \tag{2}$$

$$\min_{\forall e \in p} w_{e,0} \geq C_0 \tag{3}$$

Equation (1) expresses the selection of the optimal path from the set of non-dominated paths $P^*_{s,t}$ computed over the S inter-domain routes. Equation (2) expresses the additive resource constraints on selected path segments within the different inter-domain routes. Equation (3) shows the resource constraints on local bandwidth associated to each non-dominated path.

3.2.1 Problem Classification

The ID-MCOP-MDR problem is a MCOP problem which is classified as NP-complete (e.g.Jaffe (1984)). The only difference between our problem and the MCOP one, is the context of solving the problem, i.e. constraints that force us to solve the problem by parts (per-domain); if there were no confidentiality constraints or management restrictions between domains and if a centralized entity (that would have a global vision of all networks resources and their states) exists, it will be exactly the same as a MCOP problem in mono-domain.

3.3 Algorithm for Inter-Domain Multi-Constrained Optimal Path Over Multiple Domain Routes

This section details our proposal for a distributed algorithm, called Inter-Domain Multi-Constrained Optimal Path Over Multiple Domain Routes (ID-MCOP-MDR) algorithm. The ID-MCOP-MDR algorithm exactly computes inter-domain paths over multiple domain routes subject to m-QoS constraints and optimizes an objective function over those paths.

3.3.1 Basic Principles

The work of Van Mieghem and Kuipers on the MCP problem, in an intra-domain context Van Mieghem et al. (2004), has inspired our work on ID-MCOP-MDR algorithm, which is based on four key principles (Djarallah et al. (2011)): non-linear length function, k-shortest path storage (k-shortest paths are stored on each intermediate node), non-dominance (reduce the research space), and path segmentation (confidentiality aspects are preserved between neighbouring domains).

3.3.2 Description of the ID-MCOP-MDR Algorithm

In the following section, we detail the meta-code of the ID-MCOP-MDR algorithm. The request is to compute one or more paths from a source node s to a destination node t, subject to a constraint vector C_j, where $j = 0, ..., m$, that minimize a cost function $Z(P^*_{s,t})$. Other input parameters are recovered locally, such as the network graph of the source domain D_s and the set of neighbouring domain's path segments

$PathSegs_{i-1}$. The procedure starts initializing previous path segments to ϕ and then calls the subroutine *ID-MCOP-MDR*, described by algorithm 1, to trigger the computation over different inter-domain routes. A timer is activated until reception of potential non-dominated paths through the different inter-domain routes. Once the timer expires, subroutine *Compute_GlobalOptimumPath* performs a global optimization (in our case, minimization of $Z(P^*_{s,t})$ = minimization of $l_\infty(P^*_{s,t})$).

The subroutine *ID-MCOP-MDR* (algorithm 1) contains two different treatments: one is achieved by intermediate domains, another by the target domain D_t. Intermediate domain processing is exhibited by lines 1 to 7 of algorithm 1. It begins with the concatenation of previous path segments (initially empty) to the present network graph (line 2 of algorithm 1). Each path segment starts with the source node s and ends with a border node. The concatenation allows the computation of constrained segments, still, from the same node s to an ingress border node of a next domain. Ingress border nodes are extracted by the function *IngressNodesOfDownstreamDomains*, that uses as an input (using the function *Next*(D_i, D_t, t) in line 3) downstream domains, which are able to reach the target t. The set Ω of ingress border nodes, is then used at line 4 by subroutine *Compute_Non-dominatedPaths* to compute non-dominated path segments within the current domain, which are transmitted to neighbour domains that could reach the destination, at lines 5-7. The second part, illustrated between lines 8 and 13, differs from the first one by replacing the ingress nodes of next neighbour domains by the target node t (line 10) and by sending back the path segments to the source domain (line 12) - instead of to neighbours - in order to perform a global optimization.

Algorithm 1 ID-MCOP-MDR($G_i, s, t, D_t, C_j, PathSegs_{i-1}$)

1: **if** $D_i \neq D_t$ **then**
2: $G_i \leftarrow G_i \bigotimes PathSegs_{i-1}$ //graft temporary $PathSegs_{i-1}$ to G_i
3: $\Omega \leftarrow$ *IngressNodesOfDownstreamDomains(Next(*D_i, D_t, t*))*
4: $PathSegs_i \leftarrow$ *Compute_Non-dominatedPaths(*G_i, s, t, Ω, C_j*)*
5: **for** each domain $D^+ \in Next(D_i, D_t, t)$ **do**
6: *ForwardTo(*D^+, s, t, C_j, $PathSegs_i$*)*
7: **end for**
8: **else** $\{D_i = D_t\}$
9: $G_i \leftarrow G_i \bigotimes PathSegs_{i-1}$
10: $\Omega \leftarrow t$
11: $P^*_{s,t} \leftarrow$ *Compute_Non-dominatedPaths(*G_i, s, t, Ω, C_j*)*
12: *SendBackTo(*D_s, $P^*_{s,t}$*)*
13: **end if**

The subroutine *Compute_Non-dominatedPaths* computes intermediate path segments between the source node s and the nodes of Ω. It takes into account the requested bandwidth and other QoS constraints. We have detailed this subroutine in previous work (Djarallah et al. (2011)).

Notice that ID-MCOP-MDR does not generate loops. In each domain, loops are inherently prevented by using a loop avoidance mechanism, see section 4.3. In addition, we demonstrated (Djarallah et al. (2011)) that our algorithm has,

1) worst-case runtime complexity in order of $O(R.D.k^2.m.(k.|V'| + |E'| + |V'^2|))$, where R is the number of explored routes and D the number of domains, and 2) worst-case space complexity in order of $O(R.D.(2 + 2.k.|V'^2| + |V'|.E') + logP^*)$.

4 Multiple Path Exploration

4.1 Challenges

The exploration of different inter-domain sequences simultaneously involves challenges to consider:

- *Inter-domain PCE discovery.* In inter-domain context, several inter-domain chains could be used to reach the destination. The first challenge is to find out these chains (corresponding PCEs) in order to allow the exploration of multiple routes. The discovery of routes could be done either before the constrained path computation phase or simultaneously. In previous work (Pouyllau et al. (2009)), we proposed some solutions to identify different inter-domain routes based on technical and economic constraints at the level of the Service Plan. However, a domain may, in practice, contain multiple PCEs (for the path computation load balancing, purpose of redundancy, PCE's dedicated functions, etc.). In this case the appropriate PCE could be identified either at the Service Plan, or the Control Plan using one of PCE discovery mechanisms discussed in section 2.
- *Inter-domain loops.* The exploration of different inter-domain sequences could lead to loops. The challenge is to associate a mechanism that detects and avoids loops during the inter-domain PCE discovery process. In section 4.3 we propose a new mechanism to avoid inter-domain loops.
- *Termination.* The termination of the computation protocol is an important issue, especially because the protocol that we propose in this paper explores several paths simultaneously to reach the same destination. To ensure the termination, we propose to use a coloring mechanism (see section 4.4.1) to mark the different intermediate constrained paths. This marking helps the destination (domain/PCE) to identify the computed paths and to wait for in-progress paths. In addition, the colouring mechanism allows the synchronization of the different received paths through the explored inter-domain routes.

4.2 Overlay of Inter-Connected PCEs

One important issue in the multi-path computation is to determine to which next domain/PCE the request has to be forwarded to. Therefore, we propose a solution, at the level of PCEs that guarantees the exchange of the reachability information between PCEs in order to construct a map of inter-connected PCEs. Figure 1 shows a set of inter-connected domains where the computation feature is managed by one PCE per-domain.

An overlay network of PCEs gives the possibility to PCEs to construct, update and forward their PCE Reachability Tables (PRT). Furthermore, the PRTs allow

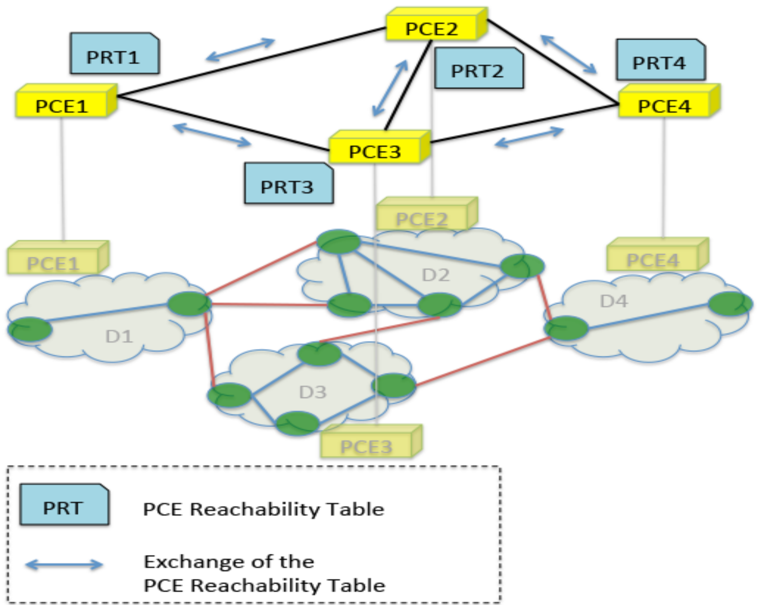

Figure 1: Overlay view

PCEs to construct an overlay topology of PCEs, and therefore identify inter-PCE paths. Notice that, PRT can be feeded directly by the Service Plan.

This technique can easily be implemented within alliances based on trust and formed between a limited number of domains. Alliances can be seen as those formed in the field of airlines companies. The reduced number of domains that form the alliance limits the broadcasting of the information about the reachability of PCEs.

4.3 Inter-Domain Loop Avoidance

In an inter-domain context with a PCE-based framework, computation path requests travel from one PCE to another until the destination. During the forwarding of the request, loops could be formed involuntary. Indeed, loops lead to under-optimal paths and generate extra-computation for the process. Here we propose a mechanism that define, detect and prevent loops within PCEP requests or another protocol that allows the request forwarding between domains.

Loop-Avoidance Mechanism. The basic idea consists in two new fields defined in the PCEP requests. The first one allows the Path Computation Client (PCC) which asks for an inter-domain path computation to specify the policies for loop detection using the following parameter:

- L_Gra = E_Gra, transmitted in PCEP Request messages, where:
  - E_Gra = {AS, domain, area} represents a level of granularity: describes the level on which loop detection should be applied: for instance "area", "AS", "domain" or any normalized level of granularity. In our case the level of granularity is "domain".

– L_Gra gives a definition of what are loops from the requester point of view.

The second one is named "D_Path" and contains an ordered list of all crossed domains identifiers (D_id). The representation can be more complex depending on the level of granularity, for instance if the domain is an Autonomous System (AS) with several areas, the D_id will be a couple of AS_id and the Area_id (e.g. (AS)2(Area)1), as depicted in figure 2. Each crossed PCE adds the "D_id" of its domain to the D_Path. Concerning the AS, this identifier might be the AS Number (ASN). Information about areas identifiers that is sent to another AS can be considered as a violation of the confidentiality rule. For this reason we propose to erase the sequence of sub-networks identifier, from the D_Path, when the request is forwarded to a high level network granularity (e.g. from the Area-level to the AS-level), as we can see it on figure 2 when the D_Path is sent from (AS)2(Area)2 to (AS)3.

A PCE, before forwarding the request, checks if the next-hop PCEs satisfies or not the **selection rule** (algorithm 2). If true, it sends the request to the authorized downstream PCEs. This rule allows for automatic loop prevention when computing inter-domain constrained paths in a distributed manner. One of the main challenge in loop prevention is also to not detect "false positive" loops. False positive loops are not "real" loops. Our proposal detects loops without detecting false positive ones in the sense of the requester policy described in L_Gra.

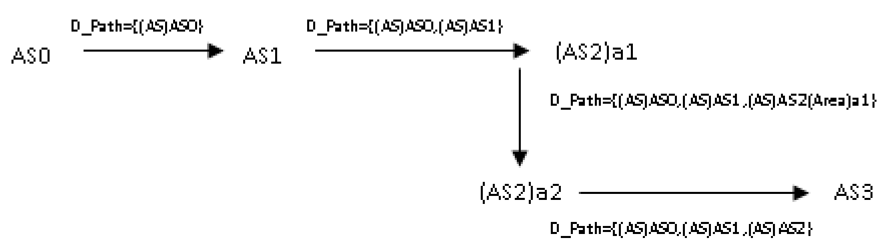

Figure 2: Example of building and forwarding of the D_Path

Rule. The current PCE_i, which has received a request, of the domain D_i checks PCE_{i+1} can be the next PCE or not.

Algorithm 2 Selection rule

If ID_id(PCE_i) and D_id(PCE_{i+1}) $\in$ D_$\text{Path}_{received}$ and E_Gra(PCE_{i+1}) E_Gra(PCE_i) $\in$ L_Gra
Such that
  D_id(PCE_i) $\neq$ D_id(PCE_{i+1})
  D_id(PCE_i) precedes D_id(PCE_i) in D_$\text{Path}_{received}$
Then
  PCE_{i+1} is not a good candidate (i.e. it creates a loop)

The current PCE analyzes its received D_Path (D_Path$_{received}$) and checks if a given PCE can be the successor and will not create a loop.

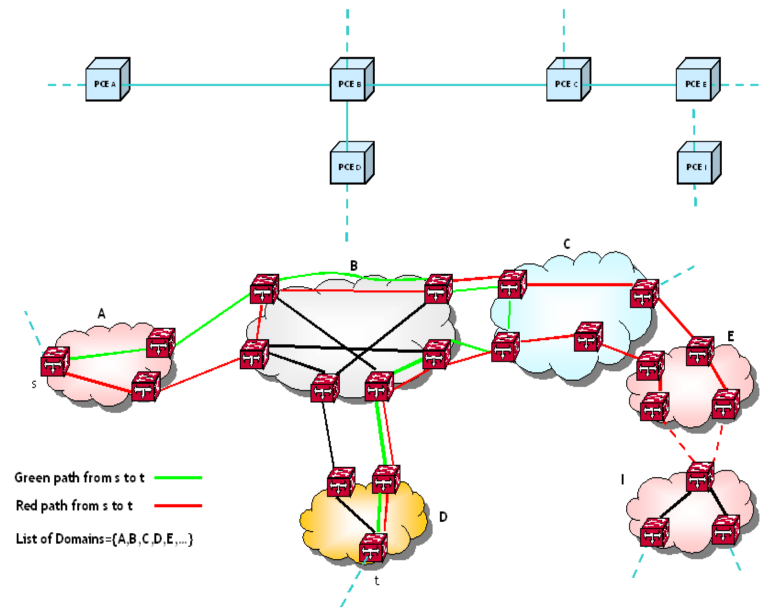

Figure 3: Example of false inter-domain loops

Note that the solution takes in consideration the presence of several PCEs in the same domain though the condition of the rule "D_id(PCE$_i$ $\neq$ D_id(PCE$_{i+1}$)".

Examples of false inter-domain loops. Figure 3 illustrates an example of multi-path establishment in an inter-domain scenario from a source node named s to a target node named t. We assume that 1) each domain has only one PCE; 2) each PCE has its proper PCE Rreachability Table; and 3) domain D cannot be reached directly through domain B (due to PCE request constraints violation for instance: the available bandwidth is not enough important on domain B intra-links at the exception of the links used by the green and the red path). At the reception of the request by PCE$_B$, initially sent by PCE$_A$ with L_Gra={domain}, it will check all the possibilities to reach D: 1) directly through its border routers with D (this case is excluded by the assumptions) and 2) indirectly through domain C. The inter-PCE routing table of PCE$_C$ indicates that to reach D, two possibilities are available: 1) through the domain B (the green path) and 2) through the domain E (the red path). If we apply the rule described above, PCE$_C$ and PCE$_E$ would forward their request respectively to PCE$_B$ and PCE$_C$. A complete messages exchange example (green and red paths) is depicted below:

Green path exchanged messages:

- PCE$_A$ $\overset{D\_Path=\{(AS)A\}}{\rightarrow}$ PCE$_B$
- PCE$_B$ $\overset{D\_Path=\{(AS)A(AS)B\}}{\rightarrow}$ PCE$_C$

- $PCE_C \overset{D\_Path=\{(AS)A(AS)B(AS)C\}}{\rightarrow} PCE_B$
- $PCE_B \overset{D\_Path=\{(AS)A(AS)B(AS)C\}}{\rightarrow} PCE_D$ because $D \notin \{ABC\}$ and as axplained above, B is **before** C in D_Path

Red path exchanged messages:

- $PCE_A \overset{D\_Path=\{(AS)A\}}{\rightarrow} PCE_B$
- $PCE_B \overset{D\_Path=\{(AS)A(AS)B\}}{\rightarrow} PCE_C$
- $PCE_C \overset{D\_Path=\{(AS)A(AS)B(AS)C\}}{\rightarrow} PCE_E$
- $PCE_E \overset{D\_Path=\{(AS)A(AS)B(AS)C(AS)E\}}{\rightarrow}$ successors PCE_E
- predecessors $PCE_E \overset{D\_Path=\{(AS)A(AS)B(AS)C(AS)E^{\star}\}}{\rightarrow} PCE_E$
- $PCE_E \overset{D\_Path=\{(AS)A(AS)B(AS)C(AS)E^{\star}(AS)E\}}{\rightarrow} PCE_C$
- $PCE_E \overset{D\_Path=\{(AS)A(AS)B(AS)C(AS)E^{\star}(AS)E(AS)C\}}{\rightarrow} PCE_B$
- $PCE_E \overset{D\_Path=\{(AS)A(AS)B(AS)C(AS)E^{\star}(AS)E(AS)C(AS)B\}}{\rightarrow} PCE_D$

The symbol * means any finite sequence of domains. From our assumption D_Path={A-B-C-B-C-B-D} is a loop, whereas D_Path={A-B-C-B-D} (green path) is not. The rule ensures that {A-B-C-B-C-B-D} is detected.

4.4 Protocol for Multi-Constrained Paths Over Several Inter-Domain Routes

4.4.1 Basic Principles

The basic idea of our procedure to compute multi-constrained paths for an MPLS-TE LSP over multiple inter-carrier routes from a source node to a destination node comprises:

- A request for an inter-carrier MPLS-TE LSP is sent by a PCC to a particular PCE (called source PCE) of its carrier, to trigger the exploration mechanism and to compute multi-constrained path(s) from a source node to a destination node. The PCC can be the source node.
- A Session ID flag is added to the PCE request to identify the same path computation process with the same couple source/destination. In this specific case (PCE-based computation) we propose to reuse the Request ID field used in the PCEP standard protocol.
- We propose to add a PCE Path field, transmitted in the PCE request message and completed (PCE address or PCE id appending) during the message transmission. The role of the PCE Path could be twofold: 1) optimization of loop detection mechanism (Section 4.3) : each PCE only enters the loop detection algorithm when it finds its address (or id) in the PCE Path of a received PCE request message and 2) in case of stateless PCEs, PCE Path allows the destination PCE to send the optimal path(s) back to the source PCE following the reverse path of the PCE Path.

- The source PCE identifies the next PCEs to be addressed using one technique of those previously explained in Section 4.1 and 4.2.
- The source PCE computes the VSPT from the source node to every entry border nodes of next PCE neighbours, involved in the computation scheme.
- To ensure the termination of the exploration protocol and the synchronization of different VSPTs, we apply different mechanisms:
  - To limit the number of hops, a new flag must be added to the PCE request messages, Time To Live (TTL). The value of this flag is initialized by the source PCE and decremented each time an intermediate PCE receives the request.
  - As the exploration concerns several routes, the PCE destination may receive multiple VSPTs. The PCE destination must wait for the VSPTs before making a decision. To avoid the infinite waiting time, a timer (Target PCE Waiting Timer or TPWT) is armed by the PCE destination once the PCE request message with a new Session ID (first VSPT related to a Session ID) is received.
  - To manage the waiting time of the timer and therefore the termination of the exploration protocol, we propose a coloring mechanism to be applied to the different VSPTs. The white color means that the node is under the scope of the PCEs which are in the PCE Path and the black one means that other trees are sent to other PCEs. Let takes a focus on three PCE neighbors; PCE_i, PCE_{i+1} and PCE^*_{i+1} as illustrated by fig.4. PCE_i computes the VSPT from s to concerned entry border nodes of the carrier-PCE_{i+1} and carrier-PCE^*_{i+1}. Before sending the VSPT to PCE_{i+1}, PCE_i applies the coloring rule on this VSPT by stamping end nodes of the VSPT: 1) Nodes that corresponds to concerned entry border nodes of carrier-PCE_{i+1} are stamped with white color, in order to indicate to PCE_{i+1} that this sub-tree (source node and white nodes) corresponds to computed constrained path segments within carrier-PCE_i. This sub-tree should be merged with the network graph of carrier-PCE_{i+1} to continue the computations, 2) the other nodes of the VSPT are stamped with black color in order to inform PCE_{i+1} that there is another VSPT sent to a second PCE capable also to reach the same destination. This stamping operations lead to construct $VSPT_i$. Similarly, $VSPT^*_i$ is constructed and sent to PCE^*_{i+1}.
- We propose to reuse the field Include Route Object (IRO) of PCE request message (carried in PCEP) in the same way that the Explicit Route Object (ERO) is used in the PCE reply message to transport the VSPT for the BRPC procedure.
- When one or more VSPTs are received by an intermediate PCE, this last one grafts the VSPT(s), according to the black and white nodes, to its graph that models the network topology of the carrier. Note that, links connecting the source node to black nodes are not concerned by the path computation.

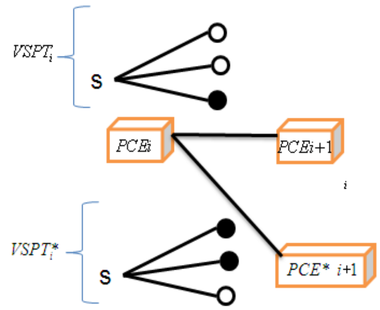

Figure 4: Coloring principle

Once the merge is done, the PCE trigger the computation process within the new network graph (from the source node to entry border nodes of the involved downstream carriers).

- When a PCE sees that a data path of the VSPT cannot further satisfy the QoS constraints, it should prune this data path without sending extra control messages to the destination PCE. The latter will know that the data path is pruned thanks to the node color, which is white in this case (uncompleted data path with white nodes means pruned data path and uncompleted data path with black nodes means actual uncompleted data path). Such a pruning in the forward direction allows for limiting the flooding of messages.
- The destination PCE uses the coloring mechanism also to synchronize the different VSPTs. The first VSPT received, indicates if more VSPTs could be received through alternative inter-carrier routes. This information is extracted from black/white nodes contained in the received VSPT. In this case the PCE destination must wait for the other VSPTs before waking a decision – complete the computation and return back the solution(s).

4.4.2 Example

To well understand, an example showing the execution of the calculation using our computation protocol is illustrated in figure 5. Four carriers (D_1, D_x, D_y, D_2) are interconnected and everyone has its own PCE (PCE$_1$, PCE$_x$, PCE$_y$, PCE$_2$). A client located on node 'A' requests, via its PCC, the optimal path from itself to the node 'Z'. This request is called PCEReq and identified with a Session ID = 'i'. The remainder of the running of this example is done in different steps, as we can see it on figure 5:

- Step 0: PCE$_1$ identifies the next PCEs that are able to reach the destination (or know other PCEs capable to reach the destination) using the PRT (PCE Reachability Tables) or other mechanism (Section 4.2). Then, PCE$_1$ computes the path segments (VSPT) from the source node 'A' to all entry border nodes of next PCE neighbors, involved in the computation scheme.
- Step 1: PCE$_1$ applies the coloring on nodes of the computed VSPT. It stamps nodes 'H' and 'I' with black color and nodes 'J' and 'K' with white color to

indicate whether a path tree node is or is not under the scope of a neighbor PCE to which a request is to be forwarded (PCE_x in this case). Then PCE_1 formulates a request PCEReq with the same Session ID (i) as the PCC's request, add itself to the PCE Path field, initialize the TTL to 100 hops and adds the colored VSPT, and send it to PCE_x.

- Step 1': This time PCE_1 stamps nodes 'H' and 'I' with white color and nodes 'J' and 'K' with black color to indicate whether a path tree node is or is not under the scope of PCE_y. Then formulates another request with the same Session ID (i), add itself to the PCE Path field, initialize the TTL to 100 hops and adds the colored VSPT, and send it to PCE_y.
- Step 2: PCE_x receives the request from PCE_1 with the new VSPT, decrements the TTL to '99', graft the received VSPT to its proper graph and compute the new VSPT from the source node 'A' to entry border node of the target carrier-PCE_2. PCE_x applies node colors rule on the expanded VSPT and formulate a new request to PCE_2 with the same Session ID, the new value of TTL, adds the identifier of PCE_x to the PCE Path and adds also the new VSPT.
- Step 3: PCE_y does the same thing as the other; formulate a new request with the new results and sent it to PCE_2.
- Step 4: PCE_2 receives first, either the request from PCE_x or from PCE_y. PCE_2 starts the timer $TPWT_i$ at the reception of the first message with Session ID = 'i'.
- Step 5: PCE_2 stops the timer $TPWT_i$ at the reception of the second request message with the same Session ID, which allows completing the previous black nodes.
- Step 6: PCE_2 graft the received VSPTs to its proper network graph and builds an aggregate path tree (VSPT) based on this new network graph from the source node to the destination node. The VSPT is complete when no black nodes left. PCE_2 completes the VSPT, then it selects and returns the optimal path(s) to PCE1 directly or through the reverse PCE path.

5 Simulation Experiments and Results

In order to evaluate feasibility and efficiency of the proposed exploration model and corresponding computation approach, we conducted several experiments on different inter-domain scenarios.

5.1 Scenarios and metrics

We evaluated the performance of the ID-MCOP-MDR algorithm and its exploration protocol using a self-written JAVA network simulator. The network topologies are generated based on a Waxman model by using the BRITE software generator. The generated topologies are depicted in figure 6. Information about the

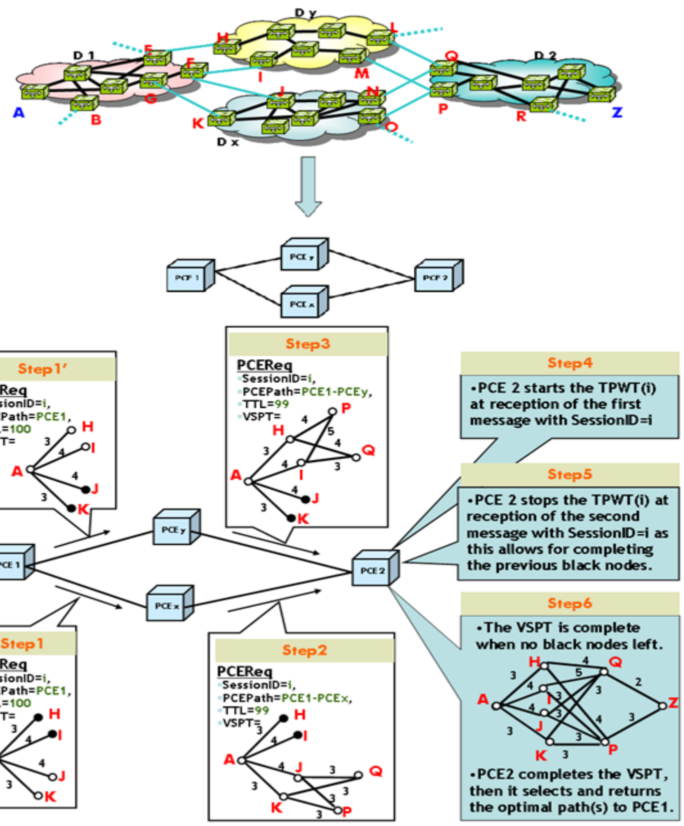

Figure 5: Example of the proposed computation protocol

number of nodes and edges per domain are summarized in figure 7. For example, topology 1 contains 5 interconnected domains with 600 network nodes and 1200 edges per-domain.

Each edge is associated with three QoS metrics (the first one is the bandwidth and the others are additive integers). Additive QoS metrics are positively correlated within $[1, 1000]$. The bandwidth capacity of each link is fixed to 10 Gbps. Throughout the simulations, requested bandwidth is set to 64 Mbps. Usually, the constraints expressed in the request are either loose (polynomial computation algorithm can compute paths) or tight. Thereafter, all generated requests are tight, but with different level of tightness: T1-to-T5, where T5 is the tightest one.

The purpose of the simulations is: 1) to study the impact of exploring multiple routes on the number of satisfied requests and the number of solutions per-request, 2) to evaluate the generated overhead messages when one or divers inter-domain routes are explored, 3) to study the impact of the exploration procedures on the number of satisfied requests, and finally 4) to evaluate the runtime of the different exploration procedures.

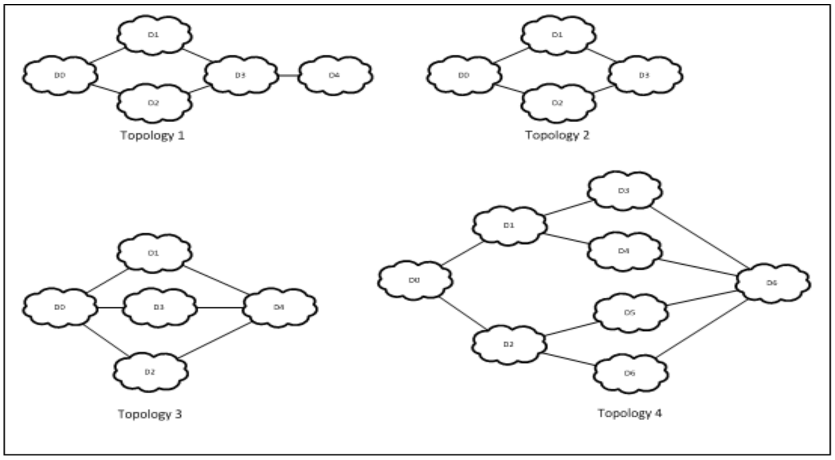

Figure 6: Reference topologies

5.2 Simulation Results

Satisfied requests & Average Message Overhead (AMO). ID-MCOP-MDR and its exploration protocol use control messages to check the availability of resources, exchange of the request, acknowledge a path computation request or terminate a request. These control messages can be considered as network overhead since they consume some network resources. Therefore, given a set of generated requests, for the same couple source/destination, we measure the average message overhead in order to find (if it exists) the constrained path(s).

In order to show the benefits of exploiting multiple paths instead of one, as is done with BRPC, we conducted some simulations on different topologies (see figure 7). For the first series of simulations, we launched the computation of all requests on a single inter-domain route using BRPC to see the number of satisfied requests and the Average Message Overhead (AMO). As the goal is to make the evaluation on a single path, for the first series, we have experienced it on each inter-domain route separately, and then we calculated the average for both the number of successful requests and the AMO. In the second series of simulations, requests follow a first inter-domain route until the depletion of network resources, then a second route is used and so on (we call it Sequential BRPC). Then, for the third series of simulations, we do the same thing as the second except that the inter-domain routes are chosen randomly. Finally, we evaluate our solution that explores multiple inter-domain routes in parallel.

In the first case (see figure 7) and with the topology 1, we had a success of 91,6 requests among the 200 requests compared to other solutions where the number of successful requests is higher. Nevertheless, these results are obvious, because the number of success request naturally increases with the increase in the number of explored inter-domain routes. This conclusion remains valid on experiences made on the other topologies. Thus we note that the number of requests is roughly divided by two or more. Contrariwise the AMO does not follow the same rule, certainly the number of AMO (obtained with 'BRPC on one inter-domain route') is lower compared to other solutions (sequential BRPC, random BRPC, and ID-MCOP-MDR), but it exceeds in most cases the half. This is due to attempts to calculate constrained path even if the network resources on one inter-domain route are no longer available.

Topologies [nodes, edges(per-domain)] [# of domains] [# of sent requests]	BRPC on one domain sequence[average number of satisfied requests] [# of AMO]	Sequential BRPC[number of satisfied requests] [# of AMO]	Random BRPC[number of satisfied requests] [# of AMO]	ID-MCOP-MDR [number of satisfied requests] [# of AMO]
Topo 1 [600,1200] [5 Domains] [200 req]	[91.60 req] [708.652 msg]	[195 req] [1048 msg]	[187 req] [990.862 msg]	[200 req] [992.772 msg]
Topo 2 [400,800] [4 Domains] [200 req]	[69.23 req] [400.846 msg]	[132 req] [686.862 msg]	[131 req] [648.826 msg]	[140 req] [629.892 msg]
Topo 3 [300,600] [5 Domains] [200 req]	[53.654 req] [302.125 msg]	[179 req] [510.056 msg]	[165 req] [6402.366 msg]	[179 req] [419.958 msg]
Topo 4 [200,400] [8 Domains] [200 req]	[38.5 req] [260.452 msg]	[188 req] [603.370 msg]	[180 req] [588.567 msg]	[192 req] [502.357 msg]

Figure 7: Satisfied requests & Average Message Overhead

Now we compare the various experiments made using BRPC several times through the different inter-domain routes, within all topologies, in a sequential or random may, and our solution that explores multiple inter-domain routes in parallel. Looking at Figure 7 we see that the number of satisfied requests using our solution is greater than (topology 1, 3 and 4) or equal (topology 3) to the number of satisfied requests when we use BRPC to explore different inter-domain routes in a sequential or random manner. This is due to the fact that requests are handled according to the different inter-domain routes, then the best constrained path is selected, therefore the reservation of network resources is done in an optimal way. However, contrary to what we expected, our solution does not generate more overhead messages (AMO) than the other two solutions. For example, with topology 1, the ID-MCOP-MDR solution has generated 992,772 overhead messages. Contrariwise, when BRPC procedure is used on several route, in this case the number of overhead messages is slightly higher (1048 messages). This is valid with all other topologies we used. This difference is due to constrained path segments that are already computed and stored (when the request passes for the first time) within some intermediate domains and reused when the same request passes through the same domain another time (another inter-domain route that includes the same domain).

Mean Execution Time (MET). In figure 8 we evaluate the runtime of the different computation/exploration methods, using the topology 1. Thus, we compare the runtime of our solution that explores multiple paths in parallel with the multiple BRPC technique (sequential BRPC) and the classical BRPC procedure running on a single predetermined inter-domain route. However, requests are generated according to several level of tightness: T1, T2, T3, T4, and

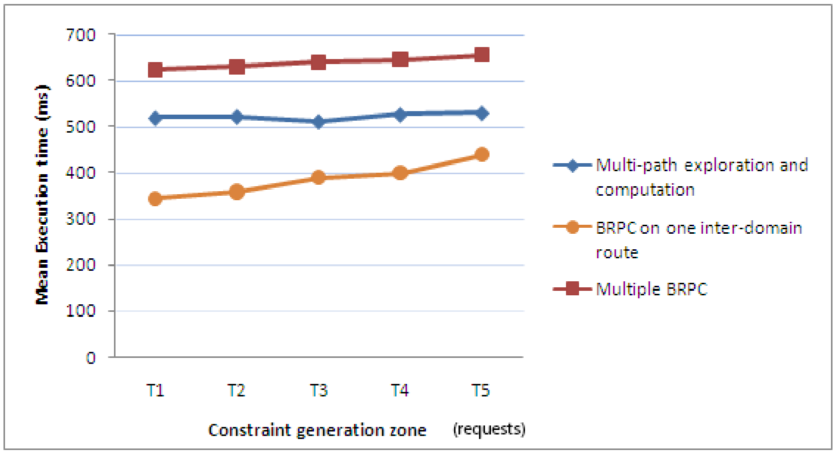

Figure 8: Runtime evaluations

T5. Therefore, we can see that the runtime related to the exploration case of one inter-domain route with BRPC is less then the runtime of the other solutions. This is justified by the simple reason that only one route is considered and after a certain number of requests the resources are no longer available, so no paths are computed, which impacts consequently the runtime of the classical BRPC procedure. Notice that the MET increases slightly with the change of the level of request tightness and therefore the increase of the success rate. The MET increases naturally with the increased number of computed paths. In figure 8, we can observe the execution time obtained by running the ID-MCOP-MDR solution which is less than the execution time of the solution that uses BRPC sequentially. This is due to constrained path segments that are stored and reused several times for the same request, over different inter-domain routes, as we explained before for the AMO.

As a conclusion, we can say that the number of feasible constrained paths increases naturally if several inter-domain routes are explored, but this also impacts the number of AMO and the runtime. Contrariwise, the ID-MCOP-MDR method can gain in terms of number of messages and runtime when intermediate domains are solicited by several inter-domain routes.

6 Conclusion

In this paper, we investigate a challenging problem in the area of inter-domain service delivery — how to improve the chances of finding end-to-end paths subject to multiple QoS constraints. Several challenges such as scalability, confidentiality, inter-domain PCE discovery, and inter-domain loops, make this problem more difficult to solve. We present a distributed inter-domain algorithm capable to compute multi-constrained paths through different inter-domain routes. Then we briefly discussed a possible solution to enable PCEs to exchange information about their ability to reach other remote PCEs, by opting for an overly-based schema. Furthermore, we presented a new mechanism to avoid inter-domain loops without excluding 'false positive loops', contrary to actual deployed solutions (e.g. loop avoidance with BGP). Finally, we proposed a novel inter-domain path computation algorithm with its protocol, which allows the exploration of various PCE chains and computes several inter-domain constrained paths according to a set of QoS requirements. Our solution provides satisfiable performance with high success rate, reasonable message overhead and runtime. These gains are particularly notable for the case where an intermediate domain belongs to several inter-domain sequences for a given request. Future studies will look at extending this work for post-path computation — resource reservation and service survivability.

References

Farrel, A., Vasseur, J-P. and Ash, J. (August 2006) 'A Path Computation Element (PCE)-Based Architecture', *RFC4655*.

Vasseur, J-P. and Le Roux, J-L. (March 2009) 'Path Computation Element (PCE) Communication Protocol (PCEP)', *RFC5440*.

Vasseur, J-P., Zhang, R., Bitar, N. and Le Roux, J-L. (April 2009) 'A Backward-Recursive PCE-Based Computation (BRPC) Procedure to Compute Shortest Constrained Inter-Domain Traffic Engineering Label Switched Paths', *RFC5441*.

Djarallah, N.B., Pouyllau, H., Le Sauze, N. and Douville, R. (June 2011) 'Business-driven PCE for Inter-Carrier QoS Connectivity Services', in FutureNetw 2011: *Future Network and MobileSummit 2011 Conference Proceedings*, Warsaw, Poland, pp.1–8.

Jaffe, J.M. (1984) 'Algorithms for finding paths with multiple constraints', *Networks: an international journal*, Vol. 18 No. 14, pp.95–116.

Sanguankotchakorn, T. and Perera, N. (2010) 'Hybrid Multi-Constrained Optimal Path QoS Routing with Inaccurate Link State', in Networks (ICN): *Proceedings of the 2010 Ninth International Conference on Networks*, Menuires, France, pp.321–326.

Rekhter, Y., Li, T. and Hares, S. (January 2006) 'A Border Gateway Protocol 4 (BGP-4)', *RFC4271*.

Vijayanand, C., Bhattacharya, S. and Kumar, P. (December 2007) 'BGP Protocol extensions for PCE Discovery across Autonomous systems', *draft-vijay-somen-pce-disco-proto-bgp-04*.

Le Roux, J-L., Vasseur, J-P., Ikejiri, Y. and Zang, R. (January 2008) 'OSPF Protocol Extensions for Path Computation Element (PCE) Discovery', *RFC5088*.

Boucadair, M. and Morand, P. (May 2005) 'Path Computation Service discovery via Border Gateway Protocol', *draft-boucadair-pce-discovery-01*.

Chen, M. (October 2006) 'Inter-AS PCE Path Sequence Autoexplore', *draft-chen-pce-interas-pce-sequence-autoexplore-00*.

King, D. and Farrel, A. (April 2011) 'The Application of the Path Computation Element Architecture to the Determination of a Sequence of Domains in MPLS and GMPLS', *draft-king-pce-hierarchy-fwk-06*.

Sheldon, T. (2001) 'McGraw-Hill's Encyclopedia of Networking and Telecommunications', ISBN: 0072120053, McGraw-Hill Professional, 3rd ed., Osborne

Kuipers, F. A., Korkmaz, T., Krunz, M. and Van Mieghem, P. (2004) 'Performance Evaluation of Constraint-Based Path Selection Algorithms', *Network, IEEE*, Vol. 18 No. 5, pp.16–23.

Ziegelmann, M. (2007) 'Constrained Shortest Paths and Related Problems - Constrained Network Optimization', ISBN:3836446332-9783836446334, VDM Verlag, Saarbrucken, Germany.

Shuchita, U. and Gaytri, D. (2010) 'Exploring Issues for QoS Based Routing Algorithms', *International Journal on Computer Science and Engineering (IJCSE)*, Vol. 02 No. 5, pp.1792–1795.

Van Mieghem, P., De Neve, H. and Kuipers, A. F. (2001) 'Hop-by-Hop Quality of Service Routing', *Computer Networks*, Vol. 37 No. 3?4, pp.407–423.

Van Mieghem, P. and Kuipers, A. F. (2004) 'Concepts of exact QoS routing algorithms', *IEEE/ACM Transactions on Networking*, Vol. 12 No. 5, pp.851–864.

Di Sorte, D. and Reali, G. (2002) 'Minimum price inter-domain routing algorithm', *Communications Letters, IEEE*, Vol. 06 No. 4, pp.165–167.

Norden, S. (2005) 'Inter-Domain Routing: Algorithms for QoS Guarantees', *Computer Networks*, Vol. 49 No. 4, pp.593–619.

Tae-Il, K., Hae-Won, J., Young Chung, M. and Seong-Il, J. (2009) 'Inter-domain Routing Based on Link State Information for End-to-End QoS Guarantee', in ICACT 2009: *Proceedings of the 11th International Conference on Advanced Communication Technology*, Gangwon-Do, South Korea, Vol. 03, pp.1729–1732.

Bertrand, G., Lahoud, S., Texier, G. and Molnar, M. (2009) 'Computation of Multi-Constrained Paths in Multi-Domain MPLS-TE Networks', in NGI 2009: *Proceedings of the 5th Euro-NGI conference on Next Generation Internet networks*, Aveiro, Portugal, pp.24–31.

Amari, W., Gadhgadhi, R., Ben Letaifa, A. and Tabbane, S. (2010) 'Path Computation with Multiple Constraints QoS in Multi-Domain', in AICCSA 2010: *Proceedings of the ACS/IEEE International Conference on Computer Systems and Applications*, Hammamet, Tunisia, pp.1–8.

Cormen, T. H., Leiserson, C. E. and Rivest, R. L. (1991) 'An Introduction to Algorithms', *Cambridge, MA: MIT Press.*

Djarallah, N. B., Le Sauze, N., Pouyllau, H., Lahoud, S. and Cousin, B. (2011) 'Distributed E2E QoS-Based Path Computation Algorithm Over Multiple Inter-Domain Routes', in 3PGCIC 2011: *Proceedings of the 6th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing*, Barcelona, Spain, pp.169–176.

Pouyllau, H. and Djarallah, N. (2009) 'Distributed ant algorithm for inter-carrier service composition', in NGI 2009: *Proceedings of the 5th Euro-NGI conference on Next Generation Internet networks*, Aveiro, Portugal, pp.1–8.

Korkmaz, T. and Krunz, M. (2001) 'Multi-constrained optimal path selection', in INFOCOM 2001: *Proceedings of the 20th Annual Joint Conference of the IEEE Computer and Communications Societies*, Anchorage, AK, USA, Vol. 02, pp.834–843.