Photon Netw Commun (2011) 21:141–151
DOI 10.1007/s11107-010-0288-5

# Enhanced domain disjoint backward recursive TE path computation for PCE-based multi-domain networks

**Guillem Hernández-Sola · Jordi Perelló · Fernando Agraz · Luis Velasco · Salvatore Spadaro · Gabriel Junyent**

Received: 28 June 2010 / Accepted: 21 August 2010 / Published online: 9 September 2010


**Abstract** The ability of computing optimal routes poses new challenges when extending it to larger multi-domain network scenarios, as the quality of these computed end-to-end inter-domain routes depends on the selection of the domain sequence to be traversed. In the scope of the Internet Engineering Task Force (IETF), the Path Computation Element (PCE) Working Group has not provided definitive solutions to address the domain sequence selection problem, being still a work in progress. To this goal, the Path Computation Flooding (PCF) approach appears as a possible extension to Backward Recursive PCE-based Computation (BRPC) to calculate optimal end-to-end inter-domain paths without requiring a pre-configured domain sequence. Nonetheless, PCF presents major scalability issues in terms of network control overhead and path computation complexity, thus pleading for more accurate domain sequence selection techniques. This paper describes two novel mechanisms to establish inter-domain paths calculating the sequence of domains to be crossed when it is not known in advance. Both procedures make a good trade-off between the control overhead introduced and the accuracy of the computed end-to-end route. The obtained simulation results show the benefits of the proposed contributions, drastically reducing the control overhead while keeping the connection blocking probability close to the optimal values.

**Keywords** Multi-domain · PCE · BRPC · Domain sequence selection

G. Hernández-Sola (✉) · J. Perelló · F. Agraz · L. Velasco · S. Spadaro · G. Junyent
Advanced Broadband Communications Center (CCABA), Universitat Politècnica de Catalunya (UPC), Jordi Girona, 1-3, 08034 Barcelona, Spain
e-mail: guillem.hernandez@tsc.upc.edu

J. Perelló
e-mail: perello@ac.upc.edu

F. Agraz
e-mail: agraz@tsc.upc.edu

L. Velasco
e-mail: lvelasco@ac.upc.edu

S. Spadaro
e-mail: spadaro@tsc.upc.edu

G. Junyent
e-mail: junyent@tsc.upc.edu

## 1 Introduction

The evolution of future broadband services demanding huge amounts of bandwidth in a dynamic manner has altered the requirements of current transport networks. While current network infrastructures are static and optimized for voice traffic, the omnipresence of the Internet in the society is requesting a transition towards data-centric flexible and cost-effective backbones. In light of this, next generation transport networks have been enhanced with a control plane entity responsible for the automatic provisioning and release of end-to-end connections over a physical transport plane. In this context, the IETF has introduced the Generalized Multi-Protocol Label Switching (GMPLS) [1] as a set of protocols that allow the provisioning of connections, referred as Traffic Engineering Label Switched Paths (TE LSPs) in GMPLS, comprising multiple switching capabilities (packet, time-division multiplexing, lambda and fibre) in an integrated way. In their current state, these protocols provide complete support for single domain network scenarios. However, their extension to large multi-domain networks is still ongoing within the IETF.

Aiming to support the inter-domain path computation function, the Path Computation Element (PCE) [2] has been proposed as a network entity able to calculate end-to-end routes with computational constraints in single and multi-domain networks. Within a single domain, the PCE computes optimal paths using its Traffic Engineering Database (TED), which is commonly populated by means of the network routing protocol (i.e., OSPF-TE in GMPLS). Nonetheless, in multi-domain network scenarios, PCEs usually lack of TE information from the other domains, for instance, due to confidentiality and scalability reasons. This prevents a single PCE from computing whole end-to-end inter-domain routes, requiring cooperation between the PCEs in every traversed domain to finally obtain them.

Looking at the standardization, two general approaches have been proposed for the inter-domain path computation, which should be initially fed with a pre-determined domain sequence (i.e., the sequence of traversed domains) from source to destination. The first approach, called Per-Domain Path Computation [3], defines a procedure where the path is computed on a per-domain basis during the signalling process. Specifically, each PCE performs the path computation for the segment of the LSP that crosses its domain, selecting the best possible route to the next one in the pre-determined domain sequence. This concatenation of locally optimal path segments likely leads to sub-optimal end-to-end routes, requiring a high number of retries (i.e., crankback attempts) in order to establish a connection.

The second approach, referred as Backward Recursive PCE-based Computation (BRPC) [4], aims at finding the optimal end-to-end path through the pre-defined domain sequence. In this case, the path computation request is forwarded domain-by-domain through the pre-determined domain sequence from the source to the destination domain PCE. Upon receipt of this message, the destination domain PCE creates a Virtual Shortest Path Tree (VSPT) of potential paths from the destination node to the border nodes that provide connectivity to the upstream domain. This VSPT is then passed domain-by-domain back to the source domain PCE, so that each intermediate PCE in the domain sequence includes its local path information between border nodes to the received VSPT. In this way, on the basis of the resulting VSPT, the source domain PCE is able to obtain the optimal end-to-end path.

Note that the selection of the domain sequence in both approaches is essential to determine the optimal end-to-end path. However, no mechanisms are so far provided for obtaining this domain sequence, remaining still a work in progress within the IETF. In [5], the Path Computation Flooding (PCF) is presented as a possible extension of BRPC to provide optimal end-to-end path computation without requiring any pre-determined domain sequence. In contrast to BRPC, the source domain PCE sends the path computation request directly to the destination PCE. Therefore, the destination PCE computes and passes its local VSPT to the PCE responsible for each adjacent domain through which the source one can be reached. In the same way, all intermediate PCEs concatenate their local path information to received VSPT, forwarding it again to all their neighbouring PCEs. This flooding procedure finishes when the source PCE receives all VSPTs from its neighbouring domains. Based on this complete collection of possible end-to-end routes, the optimal one can be selected. Although leading to optimal inter-domain routes, this procedure also presents considerable scalability problems and network overhead, being initially discarded for large multi-domain networks.

In this work, we propose two different path computation procedures to set up inter-domain connections including the calculation of the domain sequence to be traversed. To this goal, standard BRPC is extended in order to allow parallel VSPT computations over multiple end-to-end domain-disjoint sequences for the same path computation request. In this way, the optimal end-to-end inter-domain route can be computed using the information of all gathered VSPTs. As suggested in [6] and [7], our proposals do not require intra-domain TE information dissemination, matching the traditionally strict confidentiality requirements between domains. Moreover, as will be shown in the obtained results, our contributions make a good trade-off between connection blocking probability and network control overhead when compared to standard BRPC and PCF.

The rest of this paper is organized as follows. Section 2 presents related work on the topic. Section 3 describes the proposed mechanisms for PCE-based inter-domain path computation. Section 4 illustrates the performance study and discusses the obtained results. Finally, Section 5 draws up some conclusions.

## 2 Related work

Multi-domain path computation has received remarkable interest from the standardization perspective. For example, in IP networks such as the Internet, Border Gateway Protocol (BGP) has been proposed to propagate the domain sequences to be traversed. In particular, the choice of the domain sequence in BGP is based on the lowest number of traversed domains (i.e., using the shortest BGP AS_PATH [8]). In case of equal number of traversed domains, tie-breaking rules are performed (e.g., preference to first learned routes, etc.) and only one domain sequence is selected. Therefore, BGP always tries to provide the domain sequences spanning the minimum number of domain hops from source to destination. This domain sequence may be used either in Per-domain Path Computation [3] or in BRPC [4]. However, this selection does not always provide optimal inter-domain paths, as will be illustrated by means of Fig. 1.

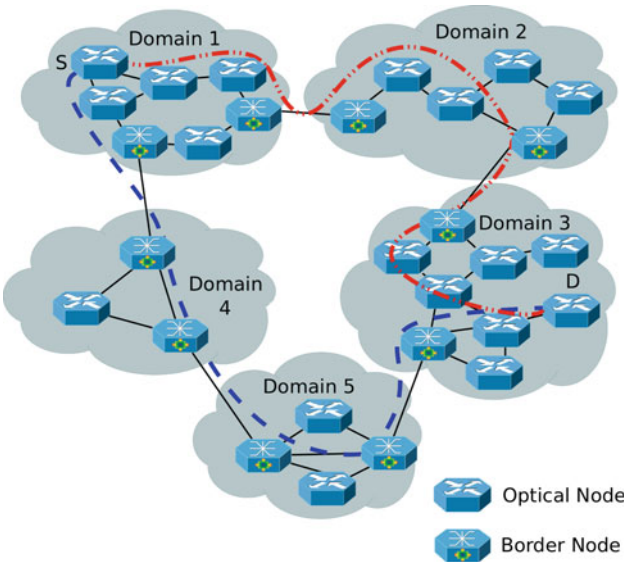


**Fig. 1** Possible end-to-end routes between S and D

In the figure, every cloud represents an independent domain with at least one responsible PCE. Let us assume that the computational entities perform BRPC to compute the end-to-end inter-domain paths using the domain sequence provided by BGP. As can be identified, two different domain sequences from S to D are possible in the network, which would also lead to different VSPTs. Assuming that the domain sequence 1–2–3 is selected as the pre-determined one, that is, the shortest one in terms of domain hops (as provided by BGP), the resulting end-to-end path would be sub-optimal, containing 13 nodes and 12 links. In fact, the optimal domain sequence from S to D becomes 1–4–5–3, even though the number of traversed domains is higher. In such a case, the end-to-end path comprises 10 nodes and 9 links. This simple example highlights that shortest domain path sequences do not always provide optimal end-to-end inter-domain routes.

We shall mention that the best case for computing optimal inter-domain paths appears when all domains share full topological and link-state information. In this way, the source domain PCE is able to compute the end-to-end inter-domain path to the destination. In line with this, the authors in [9] evaluate an algorithm with full network visibility in front of a per-domain path computation approach in a WDM multi-domain optical network. As expected, the algorithm with full network visibility reduces the blocking probability in front the per-domain approach. However, full topology dissemination is neither scalable nor matches the confidentiality requested by domain operators.

In order to enhance the inter-domain path computation without sharing complete internal domain information, topology aggregation techniques have been also proposed (e.g., see [10]). Such methods apply summarization techniques to hide the current intra-domain topology and link-state from the other domains, so that only the summarized domain information is shared among them. In this context, the authors in [10] evaluate two intra-domain aggregation methods, referred as simple node abstraction where each domain is represented as a single virtual node, and full mesh abstraction in which every domain is characterized by a group of virtual links that connect its border nodes. Moreover, they leave another aggregation method called symmetric star for further work, where all the border nodes in each domain are connected to a central virtual node. These aggregated topologies are shared amongst the different domains, finally resulting in an aggregated multi-domain topology over which the inter-domain routes are calculated. In any case, notice that these aggregation techniques provide enhanced confidentiality and scalability at expenses of sub-optimal inter-domain paths, as concluded in [11]. This is especially true when these domain topologies are highly aggregated, such as in the simple node aggregation mentioned above.

Even though topology aggregation presents better scalability for larger network scenarios, as the aggregated topologies usually comprise less information than the real ones, a considerable number of updates between domains may be required to maintain the aggregated multi-domain topology updated. In order to alleviate the control overhead due to frequent dissemination, the authors in [12] present a full mesh abstraction based on pre-reserved intra-domain resources for inter-domain connections. In this way, only those newly allocated or released inter-domain connections may trigger aggregated topology updates (i.e., modifications of intra-domain connections do not trigger updates as they are not allowed to use these resources for inter-domain transit connections). From the results, the authors conclude that this approach reduces the control overhead, but at expenses of higher intra-domain connection blocking probability due to the lower utilization of the pre-reserved resources.

Rather than relying on route computations over aggregated topologies that, as previously mentioned, may lead to sub-optimal end-to-end paths, in this work we extend the standard BRPC using a domain sequence calculation mechanism. This way, the destination domain PCE calculates $k$-shortest domain-disjoint sequences to the source domain over which different end-to-end VSPTs are gathered. Aiming to provide the desired confidentiality among domains, the exchanged VSPTs consist of non-ordered lists of path-keys [13] where every path-key represents a plain path segment from a border node to a specified destination. In this way, the explicit route inside a certain domain remains hidden throughout the path computation and signalling processes.

## 3 Proposed mechanisms

As highlighted in the previous section, a distributed PCE-based cooperative scheme to compute optimal end-to-end

routes in multi-domain scenarios while matching the arisen confidentiality requirements is still a challenging problem. This section presents two alternative route computation mechanisms named $k$-Backward Recursive PCE-based Computation ($k$-BRPC) and $k$-Backward Recursive PCE-based Computation with Load Balancing ($k$-BRPC LB). As will be detailed, both mechanisms enhance the standard BRPC letting it explore multiple domain-disjoint sequences from the destination to the source domain and, consequently, obtain the optimal end-to-end inter-domain path using the collected routing information.

In order to present $k$-BRPC and $k$-BRPC LB, the multi-domain network is modelled as a graph $G(V, E)$. This global graph joins D sub-graphs corresponding to D independent domains. In particular, the domain $G^i$ is defined as $G^i = \{V^i, E^i\}$, where $V^i$ and $E^i$ represent the sets of intra-domain nodes and links in $G^i$. Note that $V^i = \left\{v_1^i, ..., v_{\gamma^i}^i\right\}$, being $\gamma^i$ the total number of nodes in that domain and $V^i = \{V^i \subset V : V^i \cap V^j = \emptyset, \forall i \neq j\}$, where $1 \leq i, j \leq D$. In turn, $E^i = \{e(w,k) \in E : v_w^i, v_k^i \in V^i; w \neq k\}$. Specifically, $e(w,k)$ describes a bidirectional intra-domain link connecting $v_w^i$ and $v_w^i$. Inter-domain links are also bidirectional, defined as $\delta_{mn}^{ij} = \{e \in E, e(m,n) : v_m^i \in V^i, v_n^j \notin V^i \mid v_m^i \notin V^j, v_n^j \in V^j; m \neq n; i \neq j\}$, where $1 \leq m \leq \gamma^i$ and $1 \leq n \leq \gamma^j$. Each domain $G^i$ has a PCE$_i$ responsible for the computation of paths inside it.

### 3.1 $k$-Backward recursive PCE-based computation

Standard BRPC calculates end-to-end paths over a single pre-configured sequence of domains. This may lead to end-to-end sub-optimal routes if the selected domain sequence is not the most appropriate one according to the current network state. Conversely, $k$-BRPC enhances the end-to-end route calculation by means of the computation of $k$-shortest domain-disjoint sequences from the destination domain to the source one. In this way, the source domain PCE can obtain an extended set of domain-disjoint VSPTs and, using them, decide the most appropriate path between a source node $\left(v_s^i\right)$ and a destination node $\left(v_d^j\right)$. Like the standard BRPC protocol, the operation of $k$-BRPC is supported on the Path Computation Element Protocol (PCEP) [14]. Basically, the PCEP protocol allows the communication between a Path Computation Client (PCC) and a PCE, or between two different PCEs.

These interactions are implemented by means of two different types of messages, namely, *Path Computation Request* (PCReq) and *Path Computation Reply* (PCRep). Into operation, the PCReq message [14] is sent from a PCC to a PCE in order to request a new path computation (note that a PCE can also act as PCC of another PCE). In contrast, the PCRep message [14] is sent back from a PCE to a PCC in response of a PCReq message. This PCRep message can contain either the set of computed paths or a negative response if no path has been found.

Algorithm 1 depicts the path computation procedure in $k$-BRPC. When an inter-domain connection between $v_s^i$ and $v_d^j$ is requested, $v_s^i$ firstly sends a PCReq to its responsible PCE (PCE$_s$). After receiving this message, PCE$_s$ forwards this PCReq message directly to the destination domain PCE (PCE$_d$). As soon as the message reaches PCE$_d$, this one calculates $k$-shortest domain-disjoint sequences to the source domain using a multi-domain connectivity graph of the whole network. As an example, Fig. 2 (right) shows this simple graph, which can be manually created during the network bootstrap phase. Each vertex represents a domain of the network and each edge characterizes the adjacency between domains (i.e., two vertexes are connected if the domains they represent have at least one inter-domain link).

```
Input: A PCEP message
begin
    if PCReq then
        if PCReq→SourceNode ∈ G^u then
            if PCReq→destinationNode ∈ G^u then
                IntraDomainComputation(PCReq→destination
                Node);
            else
                DestPCE = findPCE(PCReq→destinationNode);
                send(PCReq,DestPCE);
        else if PCReq→destinationNode ∈ G^u then
            SourceDomain = findDomain(PCReq→SourceNode);
            KshortestRoutes(SourceDomain);
            for each upstream domain G^k do
                PCRep = CreatePCRep();
                FillDomainList(KshortestRoute^k);
                UpDom^w = findUpDom(KshortestRoute^k);
                PCRep→VSPT = ComputeVSPT(UpDom^w);
                PCE^k = findPCE(UpDom^w);
                Send(PCRep,PCE^k);
        else
            discardPCEPMessage();
    else if PCRep then
        if PCRep→SourceNode ∈ G^u then
            PCRep→VSPT = ComputeVSPT(SourceDomain);
            AddVSPTInformationToNaryTree();
        else
            NextDomain^w = CheckDomainList(PCRep);
            PCRep→VSPT = ComputeVSPT(NextDomain^w);
            PCE^w = findPCE(NextDomain^w);
            Send(PCRep, PCE^w);
    else
        discardPCEPMessage();
end
```

**Algorithm 1**: $k$-BRPC Path Computation Procedure for PCE$_i$

Note that this aggregation method does not provide information about the internal domain resources. Further details on the generation of this multi-domain connectivity graph

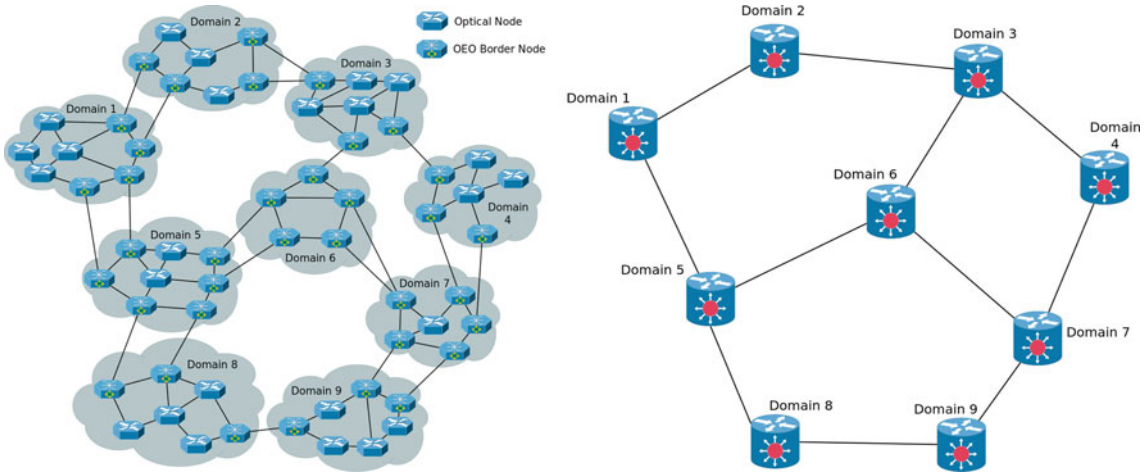


**Fig. 2** Nine-domain test network topology (*left*). Nine-domain connectivity graph (*right*)

are discussed in Sect. 3.3. In our proposed mechanism, $k$ becomes equal for all PCEs in the network. However, the maximum number of domain-disjoint sequences is tightly related to the network topology, which makes $k$ domain-disjoint sequences not attainable in certain situations.

In $k$-BRPC, these $k$ domain-disjoint sequences are treated as $k$ separate pre-configured domain sequences. Afterwards, a PCRep message is sent over each one of these pre-determined sequences. This is achieved by filling a domain list in each PCRep message with the information of the domain sequence to be explored. Therefore, $\text{PCE}_\text{d}$ also fills these PCRep messages with the VSPT to the respective upstream domain (i.e., the path segments representing the shortest route between $v_\text{d}^j$ and the border nodes connected to it). As will be detailed in Sect. 3.3, each path segment in the VSPT is assigned a cost metric and is protected using a path-key (instead of sending the explicit route), thus ensuring confidentiality among domains.

Any intermediate PCE receiving a PCRep message updates the included VSPT with its local domain route information. Right after, it forwards the message to the responsible upstream PCE according to the received domain list. This operation is repeated in every intermediate PCE until $\text{PCE}_\text{s}$ is reached. As soon as $\text{PCE}_\text{s}$ receives the $k$ different domain-disjoint VSPTs, one from each neighbouring domain, it builds an N-ary tree using all the gathered routing information. This N-ary tree is rooted at $v_\text{s}^i$ and the final leaf of each branch always is $v_\text{d}^j$. Every branch is created using a concatenation of the accumulated path segments composing the received VSPTs. After processing the N-ary tree, $\text{PCE}_\text{s}$ is able to respond the $v_\text{s}^i$ request with the optimal end-to-end path to $v_\text{d}^j$ constrained to any of the explored $k$ domain-disjoint sequences.

The usage of shortest domain sequences in terms of domain hops may lead some domains in the network to experience higher congestion than others having to allocate a larger number of inter-domain connections. This may not only impact the congested domains performance negatively, but also that of the whole multi-domain network. Aiming to achieve a uniform domain congestion and better multi-domain network performance, an enhanced version of $k$-BRPC is also proposed in the following subsection.

### 3.2 $k$-Backward recursive PCE-based computation load balancing

Similar to $k$-BRPC, in $k$-BRPC LB, $\text{PCE}_\text{s}$ sends a direct PCReq message to $\text{PCE}_\text{d}$. When this message arrives to the destination domain, $\text{PCE}_\text{d}$ computes $k$-shortest domain-disjoint sequences using the multi-domain connectivity graph, assuming these $k$ sequences as the $k$ pre-determined ones. Consequently, $\text{PCE}_\text{d}$ creates one PCRep per sequence, filled with a different domain-disjoint route to be explored in its domain list. In addition, $\text{PCE}_\text{d}$ adds to these PCRep messages its VSPT, and, after that, it sends each PCRep to the corresponding adjacent PCE.

Once receipt a PCRep message, each intermediate PCE follows the load balancing proposal detailed in Algorithm 2, firstly checking its availability to allocate an inter-domain path. Every domain presents a finite set of network resources (NR). Besides, the average number of hops to traverse a domain (ANH) is defined as the average number of steps along the shortest paths for all the possible pairs of nodes in a domain. The Dijkstra's algorithm is used for the calculation of these shortest paths. Considering a domain $G^u$ with a number $\gamma^u$ of nodes in $V^u$, $d\left(v_k^u, v_t^u\right)$ denotes the shortest

distance between $v_k^u$ and $v_t^u$, where $k \neq t$. It is assumed that $d(v_k^u, v_t^u) = 0$ when $v_k^u = v_t^u$ or $v_t^u$ cannot be reached from $v_k^u$. Then, the average number of hops is defined as:

$$\mathrm{ANH} = \frac{1}{\gamma^u(\gamma^u - 1)} \sum^{k,t} d(v_k^u, v_t^u) \tag{1}$$

Aiming to reduce the global blocking probability, each domain assumes an upper bound of the possible inter-domain LSP to be allocated. This bound, called Inter-domain Path Limit (IPL) is calculated as the ratio between NR and ANH in that given domain $\left(\mathrm{IPL} = \frac{\mathrm{NR}}{\mathrm{ANH}}\right)$. Each domain presents a different IPL due to its physical network resources and notice also that it is not distributed among the other PCEs in the multi-domain network.

**Input**: A PCRep from $PCE_{i-1}$
**Output**: A PCRep to $PCE_{i+1}$

```
begin
    if PCRep then
        if PCRep→SourceNode ∉ G^u then
            if PCRep→DestinationNode ∉ G^u then
                LSPNumber = checkCurrentLSPAllocated();
                if LSPNumber ≥ IPL then
                    IncreaseVSPTPathMetric();
end
```

**Algorithm 2**: Load Balancing Policy procedure for $\mathrm{PCE}_i$

As the number of allocated paths in a domain exceeds the IPL, the cost metric of the computed VSPT segments is multiplied by the domain network diameter. The domain network diameter is defined as the largest number of hops from one border node to another. In $k$-BRPC, each branch of the VSPT represents the shortest constrained path between two border nodes and it has associated a given cost metric. Quite the opposite in $k$-BRPC LB, when IPL is exceeded, every VSPT branch is provided selecting the worst available route connecting two border nodes assuming an increased cost metric. This assumption distributes the load of the inter-domain LSP to the less congested domains (those with the lower cost metric branches in its VSPT), aiming to reduce the blocking probability in the global multi-domain network.

The load balancing procedure is repeated in each intermediate PCE until the $\mathrm{PCE_s}$ is reached. Like in $k$-BRPC, when $\mathrm{PCE_s}$ has received $k$ domain-disjoint VSPTs from its neighbouring domains, it creates an N-ary tree using the collected routing information, and then, it provides the optimal end-to-end route to $v_s^i$ according to the current network state of the explored domains.

### 3.3 $k$-BRPC and $k$-BRPC LB in a GMPLS-controlled PCE-based network

The PCE Working Group at the IETF specifies a PCE-based architecture for the inter-domain path computation in GMPLS-controlled networks [2]. The proposed mechanisms imply considerable modifications of the PCEP client-server architecture [14] in comparison to standard BRPC, because the initial PCRep messages created by $\mathrm{PCE_d}$ are not sent as a direct and single response to a PCReq message.

For inter-domain route computation purposes, this work assumes that both the domain and PCE reachability are manually configured in the network. Nevertheless, the domain reachability, as well as the information needed to build the domain connectivity graph used for computing the $k$-shortest domain-disjoint sequences, could be also provided by means of the Border Gateway Protocol (BGP). Indeed, BGP maintains an Adj-RIB-In Route Information Base containing the Network Layer Reachability Information (NRLI) received from each neighbouring domain [8], which is enough to construct the domain connectivity graph. Concerning the PCE discovery, there are also ongoing efforts within the IETF on this subject [15].

As detailed in Sects. 3.1 and 3.2, a domain list is included in PCRep messages in $k$-BRPC and $k$-BRPC LB to indicate the sequence of domains to be explored. Such a list can be implemented using the Include Route Object (IRO) sub-object [2]. Specifically, the non-obligatory IRO sub-object in PCReq and PCRep messages is used to specify that a computed route must traverse a group of specific domains. As this sequence is not pre-defined in $k$-BRPC and $k$-BRPC LB, the IRO would remain empty in the PCReq message from $\mathrm{PCE_s}$ to $\mathrm{PCE_d}$. Then, it would be filled with the specific domain-disjoint sequence in the PCRep messages from $\mathrm{PCE_d}$ to $\mathrm{PCE_s}$.

In both mechanisms, PCEs calculate their local VSPT, which can be composed of several constrained routes. Each VSPT branch is hidden using a path-key to maintain intra-domain topology confidentiality. This path-key is transported instead of the Explicit Route Object (ERO) as a Path-Key Sub-object (PKS) in the PCRep message [13,14]. For instance, a certain path segment in a VSPT could be identified as $(\mathrm{BN}_u^j, v_u^j, C_u, K_u)$, where $\mathrm{BN}_u^j$ is the border node entry, $v_u^j$ is the specified destination, $C_u$ represents the cost metric and $K_u$ is the path-key stored in the specific PCE together with the computed intra-domain route from $\mathrm{BN}_u^j$ to $v_u^j$.

Finally, mention that additional extensions would be required to make $k$-BRPC and $k$-BRPC LB applicable to multi-carrier optical networks, without electrical termination or wavelength conversion capabilities in the domain border nodes. Each carrier's optical network is a separate administrative domain and cooperation between different carriers becomes essential in multi-domain scenarios where the wavelength continuity constraint must be preserved

throughout the whole end-to-end path. Nonetheless, the computed VSPT does not contain any information concerning the available wavelengths in the composing route segments, as seen in the paragraph above. Hence, if $k$-BRPC or $k$-BRPC LB has to be applied in transparent optical networks, PCEP extensions such as the ones proposed in [16] would have to be considered.

## 4 Performance study and discussion

The performance of $k$-BRPC and $k$-BRPC LB has been compared to standard BRPC and PCF by the use of a OMNeT++-based [17] simulator describing the 9-domain transport network depicted in Fig. 2 (left). The network topology is composed of 61 nodes and 95 links (19 inter-domain) carrying each one 8 bidirectional wavelengths per link. In this multi-carrier topology, border nodes make use of optical-electrical-optical (OEO) conversion while intra-domain nodes are all-optical (without OEO conversion). In all simulations, $10^5$ Poisson connection requests are generated following a 70/30% intra/inter-domain ratio [10]. Source and destination nodes are randomly selected for intra-domain connections and all requests demand a whole wavelength capacity. For inter-domain connections, source and destination domains are uniformly selected and source/destination nodes are uniformly chosen in their given domains. Mean holding time (HT) is set to 600 s and request inter-arrival time (IAT) varies with the network offered load. Both times are exponentially distributed. Figure 2 (right) describes the multi-domain connectivity graph over which the set of $k$-shortest domain-disjoint sequence is computed.

Figure 3 (left) depicts the connection blocking probability ($B_p$) achieved by standard BRPC, PCF as detailed in [5] and $k$-BRPC, with $k = 1, 2, 3$. Besides, it is assumed throughout the section that the shortest domain sequence provided by BGP in terms of domain hops is used as the pre-defined one in standard BRPC.

Looking at the figure, the mechanisms leading to the best and worst performance in terms of $B_p$ are PCF and standard BRPC, respectively. As introduced before, the flooding applied in PCF allows considering any domain sequence from the destination to the source node. In this way, $PCE_s$ is provided with complete information to select the optimal end-to-end inter-domain route. However, as will be numerically quantified in Fig. 5, this operation becomes unscalable as the network size increases. Contrarily, only a single domain sequence is explored in standard BRPC. This leads to suboptimal end-to-end routes in most occasions, given that the shortest domain sequence provided by BGP does not assure the optimal end-to-end path. Hence, domain sequences with a larger number of domain hops are sometimes more appropriate than the shortest ones. As previously commented in Sect. 2, the size and the current network congestion of the composing domains in the sequence significantly affect to the optimality of the computed end-to-end path. Focusing now on $k$-BRPC, it lays between standard BRPC and PFC. As seen, $k = 1$ results in the same performance as standard BRPC, since only the shortest domain sequence is explored. Nonetheless, more information can be gathered for the final inter-domain path computation as $k$ increases, improving the route selection while keeping the number of explored domain sequences bounded.

In both mechanisms, $k$ represents the upper bound of the set of VSPTs provided to the $PCE_s$ over which the inter-domain end-to-end route is computed. Additionally, it also represents the maximum number of domain sequences that can be explored between destination and the source domain. The best $k$ value is closely related to the physical topology of the multi-domain network. Specifically, the number of adjacent domains restricts the number of possible explored domain-disjoint sequences and at the same time it also limits the number of possible end-to-end routes. We demonstrate via extensive simulations (not shown in the figures) that $k$ value higher than 3 do not decrease the $B_p$ in our simulated network topology. Therefore, from now on, we select an upper bound of $k = 3$ which makes the best trade-off between $B_p$ and the introduced overhead and path computation complexity compared to PCF. To further illustrate this statement, Fig. 5 compares $k$-BRPC ($k = 1, 2, 3$) with the standard BRPC and PCF.

Figure 3 (center) shows the $B_p$ performed in each domain by $k$-BRPC and $k$-BRPC LB, with a $k = 3$ and for a fixed offered load of 180 Erlang. As shown, in 3-BRPC LB, the different domains present similar values of $B_p$. This can be attributed to the selection of the domain path to be used, which is not always the shortest one. In contrast, 3-BRPC always selects the shortest domain path and this selection drives to overload those domains that compose the shortest sequences.

As an example, Fig. 3 (center) depicts that domain 6 presents the worst $B_p$ in 3-BRPC, due to its intra-domain topology (only composed by border nodes) in addition to being situated in the middle of the multi-domain network. Conversely, domain 1 reaches the lowest $B_p$ at expenses of avoiding being transit domain in several inter-domain paths. Both abovementioned examples reflect that a poorer inter-domain LSP distribution affects the intra-domain $B_p$ as well as it also influences the global $B_p$. At this point, it seems reasonable to consider 3-BRPC LB as a feasible solution to be compared to PCF in terms of $B_p$.

The $B_p$ performed by 3-BRPC, 3-BRPC LB, and PCF is depicted in Fig. 3 (right). As described in the figure, 3-BRPC LB reduces the $B_p$ in front of 3-BRPC, but both mechanisms are outperformed by PCF, due to its end-to-end route computation procedure. As previously mentioned, PCF acquires the

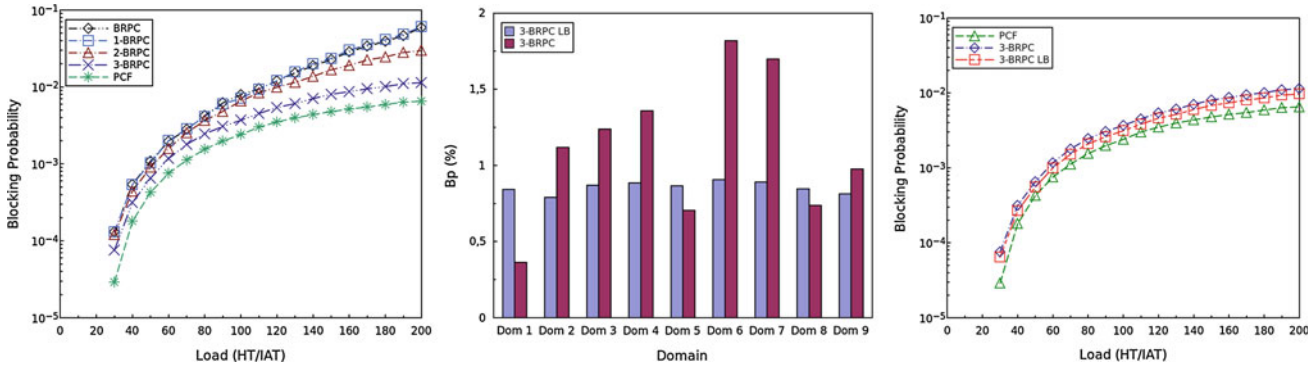


**Fig. 3** Connection blocking probability: global (*left*); distributed among domains (*center*); using load balancing (*right*)

whole set of VSPTs, not just the domain-disjoint ones, allowing $PCE_s$ to compute the optimal end-to-end route. However, to provide this complete set of end-to-end routes, PCF introduces a higher number of control messages to the network. Furthermore, due to its flooding nature, PCF also presents an increment of the path computation complexity to calculate an end-to-end path. Figure 3 (right) also depicts for low-load conditions a small $B_p$ difference among the path computation procedures, meanwhile in heavy-load conditions PCF reduces its $B_p$ around a 30% compared to 3-BRPC LB. Besides, for an offered load of 200 Erlang, 3-BRPC LB improves the $B_p$ performance of 3-BRPC around a 15%. This is achieved by means of the distribution of the inter-domain LSP to slightly larger domain sequences, driving 3-BRPC LB to a reduction of the global $B_p$ figure.

To further study the impact of the load balancing scheme on the connection blocking performance, the relative $B_p$ improvement between $k$-BRPC and $k$-BRPC LB is plotted in Fig. 4 (left), assuming $k = 1, 2, 3$ and a fixed offered load of 180 Erlang. Here, for $k = 1$ is only selected one shortest pre-determined domain sequence. As described in Sect. 3.2, the load balancing scheme is based on the choice of the less congested domain sequences to be traversed by the computed end-to-end paths. In 1-BRPC and BRPC, $PCE_s$ is only provided with a single VSPT using one pre-defined domain sequence over which an end-to-end route has to be computed. Thus, the pre-determined domain sequence as well as a single VSPT hampers the load balancing performance.

In contrast, 2-BRPC LB drastically outperforms 2-BRPC in terms of $B_p$ due to an important reason. The load balancing policy is applied independently in each domain in the selected sequence, distributing more efficiently the allocated inter-domain paths in the network. This distribution reduces the global connection blocking probability using the less congested domain sequences. As depicted in Fig. 4 (left), the relative improvement in $k = 2$ is around a 37%. In this line for $k = 3$, the distribution of the inter-domain LSPs does not sacrifice, as much as for $k = 2$, the shortest domain-disjoint sequences. This is due to the additional VSPT provided in $k = 3$ that reduces considerably the $B_p$, driving the load balancing scheme to a lower relative improvement, around 15%.

For a more complete comparison, the complementary cumulative distribution function (CCDF) of the number of hops for the allocated end-to-end LSPs in the multi-domain network is depicted in Fig. 4 (right). As expected, PCF presents the shortest end-to-end routes due to its route computation procedure. However, PCF is closely followed by 3-BRPC along with 3-BRPC LB. Figure 4 (right) depicts that PCF averages 6.30 hops per allocated inter-domain LSP which represents the lowest achieved value. Nonetheless, 3-BRPC averages 6.53 hops per allocated inter-domain LSP, along with 3-BRPC LB which averages 7.04. At this point, it has to be highlighted that 3-BRPC LB increases the number of hops in its end-to-end routes around 10% and around a 7% compared to PCF and 3-BRPC, correspondingly. These differences can be attributed to two main reasons. The first one is related to the gathered routing information which significantly affects the accuracy of the end-to-end routes. Notice that PCF acquires the total number of possible VSPTs in front a maximum of three VSPTs in the cases of 3-BRPC and 3-BRPC LB. The second reason is that 3-BRPC LB may choose a longer domain sequence to be traversed, given that it does not always select the shortest domain sequences. As previously mentioned, this selection creates a more reasonable distribution of the transit inter-domain LSPs in the network.

The shortest end-to-end routes performed by PCF are obtained at expenses of higher control overhead (in terms of PCEP messages flooded to the other PCEs) and path computation complexity to process the N-ary tree. To validate the first statement, the number of PCEP messages is evaluated for each computation request. Figure 5 (left) shows the average number of PCEP messages per path computation request delivered to the network by BRPC, PCF, and $k$-BRPC with $k = 1, 2, 3$.

As can be seen, 1-BRPC halves BRPC in terms of the average number of PCEP messages per path computation request. This difference is mainly attributed to the direct

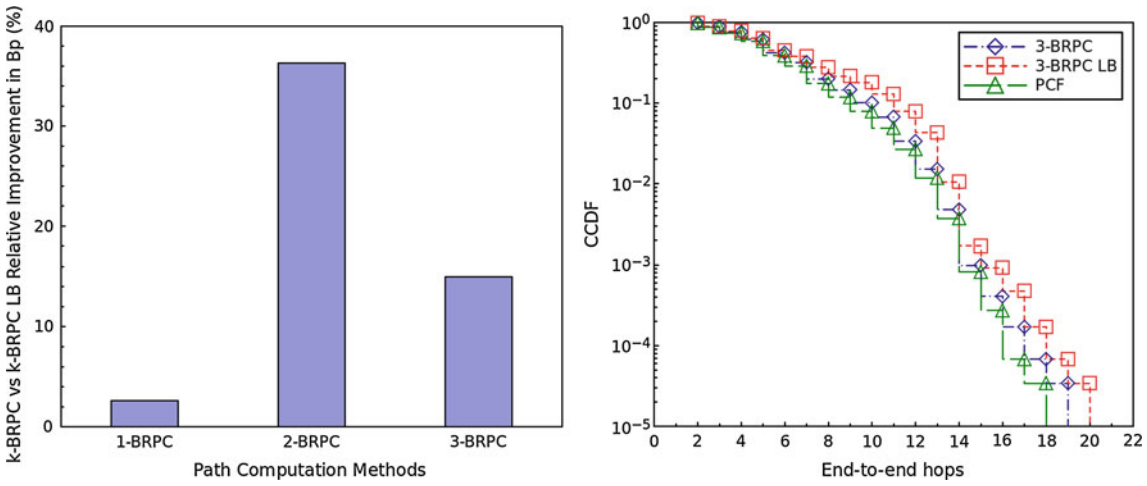


**Fig. 4** Relative blocking probability improvement between $k$-BRPC and $k$-BRPC LB path computation methods (*left*); CCDF of end-to-end hops for 3-BRPC, 3-BRPC LB, and PCF (*right*)

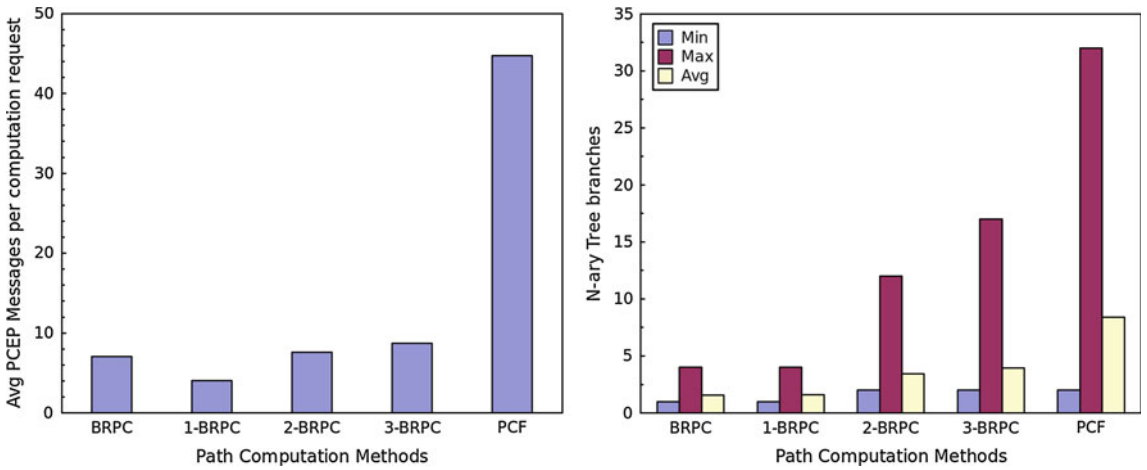


**Fig. 5** Average PCEP messages per inter-domain computation request (*left*); path computation complexity (*right*)

PCReq message that 1-BRPC sends to $PCE_d$. In addition, the global number of PCEP messages introduced in the network by 1-BRPC (not shown in the figures) is reduced around a 175% in comparison to standard BRPC. However, PCEP protocol is by nature client-server based and, as mentioned Sect. 3.3, 1-BRPC breaks this architecture sending response message without having received an explicit request.

As expected, the average number of PCEP messages in 2-BRPC is doubled in front of 1-BRPC. It is also interesting to highlight that for $k = 2$, this direct PCReq to $PCE_s$ does not significantly reduce the total amount of PCEP messages compared to BRPC. As depicted in Fig. 5 (left), 2-BRPC averages 7.6 PCEP messages per request and standard BRPC averages around 7 PCEP messages per request. Both aforementioned path computation methods average a similar quantity of PCEP messages per request while 2-BRPC halves the standard BRPC $B_p$ performance. Notice also that the average number of PCEP messages in 3-BRPC is expected to be multiplied by 3 compared to 1-BRPC, but it is only increased around 13% in comparison to 2-BRPC (averaging 8.7 PCEP messages per request). This increment is limited by the number of available domain-disjoint sequences that the multi-domain connectivity graph used in the simulations provides.

As plotted in Fig. 5 (left), PCF presents an average number of 45 PCEP messages per computation request whereas it dramatically increases around a 500% the average number of PCEP in comparison to 3-BRPC. Such a high increment hampers the scalability of PCF as well as limits its deployment in larger networks. Additionally, PCF drastically increases the path computation complexity due to the acquisition of the whole set of possible end-to-end routes in the network. This larger set of end-to-end routes increases the size of the N-ary tree to be processed by PCEs incrementing the computational load to select the optimal path from the collected VSPTs.

Figure 5 (right) compares the network scalability of BRPC, PCF and $k$-BRPC with $k = 1, 2, 3$ in terms of the path computation complexity at $PCE_s$. To this goal, at 200 Erlang ($B_p = 1\%$), the number of branches in the N-ary tree has been studied during the whole simulation interval. As expected, it shall be noted that 1-BRPC, averaging 1.7 branches per request, presents the same path computation complexity compared to BRPC which also averages 1.7 branches per request. Additionally, these two procedures compute inter-domain paths using a very limited set of the possible end-to-end routes in the network. Note that the minimum number of branches either in 1-BRPC or BRPC is 1 while the maximum number is four for both mechanisms.

Unsurprisingly, 2-BRPC doubles the path complexity of 1-BRPC and standard BRPC. Specifically, 2-BRPC averages 3.42 branches presenting 12 end-to-end routes as a maximum. The path computation complexity of 3-BRPC also shows a slight increment around 14% in comparison to 2-BRPC. It can be expected that 3-BRPC may triplicate the computation complexity, like in the average of PCEP messages per request, but the network topology limits this increment.

As seen in Fig. 5 (right), the highest number of VSPT branches is performed by PCF, yielding to a maximum of 32 branches, in some cases being very far from the other path computation methods (averaging 8.41 N-ary tree branches in the whole simulation interval). In contrast, when 3-BRPC is applied in the network, this number of branches is reduced around 50% compared to PCF, averaging 3.94 branches in the considered simulation. As a conclusion from the results $k$-BRPC and $k$-BRPC LB outperform PCF in terms of network control overhead and path computation complexity.

## 5 Conclusion and future work

In this paper, we proposed two novel procedures to compute inter-domain routes based on a calculation of the sequence of domains to be traversed. Both mechanisms enhance the BRPC path computation procedure allowing concurrent VSPT computations using different end-to-end domain-disjoint sequences, not only over the pre-configured one, for the same route calculation request.

We demonstrated via simulations that $k$-BRPC and $k$-BRPC LB drastically reduce the $B_p$ in comparison to standard BRPC. Additionally, when $k$-BRPC LB is deployed, the global $B_p$ is reduced and the domains in the network present similar intra-domain $B_p$ values. This load balancing mechanism avoids overloading the most common routes as well as the preferred domain sequences, selecting slightly larger end-to-end paths. Furthermore, both proposed procedures drastically reduce the network overhead compared to PCF. Nonetheless, the overhead reduction is achieved at the expenses of breaking the PCEP client-server model, because the opening PCRep messages are not sent as an explicit response to one PCReq. Notice also that PCF also violates this client-server PCEP architecture.

Path computation complexity is also drastically reduced in comparison to PCF. The selection of the $k$ bounds the set of possible end-to-end routes and also limits the size of the N-ary tree to be processed by $PCE_s$. This situation drives the computational load to be reduced as soon as the optimal path is selected from the gathered VSPTs. Finally, $k$-BRPC and $k$-BRPC LB presents a suitable solution to calculate end-to-end paths in a multi-domain network, making a trade-off between introduced overhead, intra/inter-domain connection blocking probability, path computation complexity, and network scalability.

Future efforts will be dedicated to apply the hierarchical PCE architecture according to the requirements of our proposed mechanisms. The main objective is to provide even better solutions in terms of connection blocking probability, routing scalability and confidentiality assurance between domains. Furthermore, the extensions to current standard GMPLS protocols required to deploy the proposed mechanisms using the hierarchical PCE architecture will be also studied.

**Acknowledgment** The work reported in this paper has been partially supported by the UPC through a FPI-UPC research scholarship grant and the Spanish Science Ministry through Project "Engineering Next Generation Optical Transport Networks (ENGINE)", (TEC2008-02634).

## References

[1] Mannie, E.: Generalized multi-protocol label switching (GMPLS) architecture, RFC 3945, October 2004
[2] Farrel, A., Vasseur, J.P., Ash, J.: A path computation element (PCE)-based architecture, RFC 4655, August 2006
[3] Vasseur, J.P., Ayyangar, A., Zhang, R.: A per-domain path computation method for establishing inter-domain traffic engineering (TE) label switched paths (LSPs), RFC 5152, February 2008
[4] Vasseur, J.P., Zhang, R., Bitar, N., Le Roux, J.L.: A backward-recursive PCE-based computation (BRPC) procedure to compute shortest constrained inter-domain traffic engineering label switched paths, RFC 5441, April 2009
[5] King, D., Farrel, A.: The application of the PCE architecture to the determination of a sequence of domains in MPLS & GMPLS, IETF draft draft-king-pce-hierarchy-fwk-04.txt, July 2010
[6] Le Roux, J.-L., Vasseur, J.-P., Boyle, J. (eds.): Requirements for inter-area MPLS traffic engineering, RFC 4105, June 2005
[7] Zhang, R., Vasseur, J.-P. (eds.): MPLS inter-autonomous system (AS) traffic engineering (TE) requirements, RFC 4216, November 2005
[8] Rekhter, Y., Li, T.: A border gateway protocol 4 (BGP-4), RFC 4271 (2006)
[9] Saad, T., Mouftah, H.T.: Inter-domain wavelength routing in optical WDM networks. In: IEEE Networks 2004, June 2004
[10] Liu, Q., Kok, M.A., Ghani, N., Muthalaly, V.M., Wang, M.: Hierarchical inter-domain routing in optical DWDM networks. In:

INFOCOM 2006, 25th IEEE International Conference on Computer Communications Proceedings, pp. 1–5, 23–29 April 2006
[11] Wan, X., Chen, Y., Zhang, H., Zheng, X.: Dynamic domain-sequencing scheme for inter-domain path computation in WDM networks. Proc. SPIE **7633**, 76330X (2009). doi:10.1117/12.851332
[12] Chamania, M., Chen, X., Jukan, A., Rambach, F., Hoffmann, M.: An adaptive inter-domain PCE framework to improve resource utilization and reduce inter-domain signaling. Opt. Switch. Netw. **6**(4), 259–267 (2009)
[13] Bradford, R., Vasseur, J.P., Farrel, A.: Preserving topology confidentiality in inter domain path computation using a key based mechanism, RFC 5520, April 2009
[14] Vasseur, J.P., Le Roux, J.L.: Path computation element (PCE) communication protocol (PCEP), RFC 5440, March 2009
[15] Le Roux, J.L.: Requirements for path computation element (PCE) discovery, RFC 4674, October 2006
[16] Casellas, R., Martínez, R., Muñoz, R., Gunreben, S.: Enhanced backwards recursive path computation for multi-area wavelength switched optical networks under wavelength continuity constraint. J. Opt. Commun. Netw. **1**, A180–A193 (2009)
[17] OMNeT++, http://www.omnetpp.org/

## Author Biographies

**Guillem Hernández-Sola** received his M.Sc. degree in telecommunications engineering in 2008 from the Universitat Politècnica de Catalunya (UPC). Currently, he is pursuing his PhD in the Signal Theory and Communications Department (TSC) at UPC. He joined the Optical Communications Group (GCO) in 2008. He is working in several R&D National and IST FP-7 European research projects like EU DICONET and VISION. His research interests concern multi-domain path computation procedures, PCE architectures in GMPLS, quality of service issues, and future IT services of next-generation optical transport networks.

**Jordi Perelló** received his M.Sc. and PhD degrees in telecommunications engineering in 2005 and in 2009, respectively, both from the Universitat Politècnica de Catalunya (UPC). Currently, he is an Assistant Professor in the Computer Architecture Department (DAC) at UPC. He has participated in various IST FP-6 and FP-7 European research projects such as EU DICONET, BONE, IST NOBEL 2, e-Photon/ONe+, and COST Action 291. His research interests concern resource management, quality of service issues, and survivability of next-generation optical transport networks.

**Fernando Agraz** received his M.Sc. degree in computer engineering in 2005 from the Polytechnic University of Catalonia (UPC). Since 2005 he has been working as a research engineer in the Optical Communications Group (GCO) at UPC, also preparing his PhD. He has also participated in various European research projects such as IST Nobel Phase 2 or E-Photon/ONe+. His current research focuses on network management and routing in GMPLS-based networks.

**Luis Velasco** received the B.Sc. degree in telecommunications engineering from Universidad Politècnica de Madrid (UPM) in 1989, the M.Sc. degree in physics from Universidad Complutense de Madrid (UCM) in 1993, and the PhD degree from Universitat Politècnica de Catalunya (UPC) in 2009. In 1989 he joined Telefónica of Spain and was involved in the specifications and first office application of the Telefónica SDH transport network. In 2003 he joined UPC, where currently he is an Assistant Professor in the Computer Architecture Department (DAC) and a Researcher in the Optical Communications group (GCO) and the Advanced Broadband Communications Center (CCABA). His interests include signaling, routing, and resilience mechanisms in ASON/GMPLS-based networks.

**Salvatore Spadaro** received the M.Sc. (2000) and the PhD (2005) degrees in telecommunications engineering from Universitat Politècnica de Catalunya (UPC). He also received the Dr.Ing. degree in electrical engineering from Politecnico di Torino (2000). He is currently an Associate Professor in the Optical Communications Group of the Signal Theory and Communications Department of UPC. Since 2000 he has been a staff member of the Advanced Broadband Communications Center (CCABA) of UPC, and he is currently participating in the DICONET and BONE FP7 EU projects. He has coauthored about 80 papers in international journals and conferences. His research interests are in the field of all-optical networks with emphasis on traffic engineering and resilience.

**Gabriel Junyent** is a telecommunications engineer (Universidad Politécnica de Madrid, UPM, 1973) and holds a PhD degree in communications (UPC, 1979). He has been a Teaching Assistant (UPC, 1973–1977), Adjunct Professor (UPC, 1977–1983), Associate Professor (UPC, 1983–1985), and Professor (UPC, 1985–1989) and has been a Full Professor since 1989. In the past 15 years he has participated in more than 30 national and international R&D projects and has published more than 30 journal papers and book chapters and 100 conference papers.