Computer Communications 33 (2010) 2215–2223

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

Enhanced crankback signaling for multi-domain IP/MPLS networks

M. Esmaeili [a], F. Xu [a], M. Peng [b], N. Ghani [a,*], A. Gumaste [c], J. Finochietto [d]

[a] University of New Mexico, ECE Department, MSC01 1100, Albuquerque, NM 87131-001, United States
[b] School of Computer, Wuhan University, Wuhan, Hubei 430072, China
[c] CSE Department, IIT Bombay, Powai, Mumbai 400 076, India
[d] Electronic Department, Universidad Nacional de Cordoba, 5000 Cordoba, Argentina

ARTICLE INFO

Article history:
Received 5 December 2009
Received in revised form 29 July 2010
Accepted 6 August 2010
Available online 18 August 2010

Keywords:
Crankback signaling
Multi-domain networks
Inter-domain path computation

ABSTRACT

Multi-domain *traffic engineering* is a very challenging problem area and crankback signaling offers a very promising solutions framework herein. Although some initial crankback studies have been done, there is still significant latitude for improving multi-domain crankback performance. Along these lines, this paper studies realistic IP/MPLS multi-domain networks and proposes a novel solution for joint intra/inter-domain signaling crankback. Namely, dynamic intra-domain link-state routing information is coupled with available inter-domain path/distance vector routing state to improve the overall search process. Mechanisms are also introduced to limit crankback overheads and delays. The performance of the proposed solution is then analyzed using simulation and compared against hierarchical inter-domain routing strategies as well as another crankback scheme.

1. Introduction

Traffic engineering (TE) in IP-based multi-protocol label switching (MPLS) and optical generalized MPLS (GMPLS) networks is a very well-studied problem area. Here a wide range of constraint-based routing solutions have been proposed, but most have focused on single "domain" settings in which a provisioning entity has complete "network-wide" topology/resource views, e.g., single autonomous system (AS) running link-state routing [1,2]. However, as user application demands grow, there is a strong desire to achieve TE provisioning across multiple domains, i.e., inter-AS TE, particularly for higher-end applications such as voice over IP (VoIP), packet video transport, virtual private network (VPN) extension, etc. Owing to obvious scalability and confidentiality concerns here, it is clear that this must be achieved in a distributed, decentralized manner.

To address these challenges, a diverse set of provisions have emerged to help improve multi-domain TE support, both at the IP/MPLS and underlying optical GMPLS layers [1–5]. On the standards side, many ubiquitous routing protocols already provide varying levels of inter-domain visibility, e.g., next-hop/path-vector dissemination in exterior gateway protocol (EGP) and hierarchical link-state dissemination in two-level open-shortest-path-first (OSPF-TE). Furthermore, the new Internet Engineering Task Force (IETF) path computation element (PCE) [3] framework also defines a comprehensive framework for multi-domain path computation and TE. Meanwhile on the research side, a host of multi-domain TE schemes have been studied. A key focus here is to address the tradeoff between inter-domain visibility and control plane complexity (i.e., dissemination, computation) [1]. For example, some have developed hierarchical link-state routing solutions to increase inter-domain visibility [6–13]. The major contributions here are graph-theoretic topology abstractions for compressing domain level state in IP/MPLS [6–9] and optical dense wavelength division multiplexing (DWDM) networks [10–13]. However, even though hierarchical routing delivers good blocking performance, associated routing overheads are very high. Hence these schemes will likely be problematic in real-world settings where carriers tend to prefer EGP distance/path-vector protocols, e.g., border gateway protocol (BGP) variants. These protocol types only provide next-hop domain and end-point reachability state and most operational versions do not support any *quality-of-service* (QoS) parameters, e.g., delay, bandwidth, etc. Hence to address these concerns, alternate "per-domain" computation schemes have been proposed for multi-domain TE [14–16], leveraging crankback signaling to overcome lower inter-domain visibility. However, many of these schemes pursue more basic "exhaustive" search methodologies (and hence entail significant signaling overheads [15]) and do not detail a structured "next-hop" strategy for crankback path computation across domain boundaries. In addition, these solutions have also not been gauged against alternate hierarchical inter-domain routing strategies.

In light of the above, there is a clear need (and significant scope) to develop more advanced multi-domain crankback solutions and gauge their performance against "global" hierarchical routing

* Corresponding author. Tel.: +1 505 277 2436; fax: +1 505 277 1439.
E-mail address: nghani@ieee.org (N. Ghani).

0140-3664/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2010.08.003

schemes. Along these lines, this paper proposes a novel enhanced crankback solution for multi-domain networks based upon the standard resource reservation (RSVP-TE) protocol. Note that this solution extends upon our recently published work in [18] by presenting more detailed discussions of the proposed scheme as well as wider range of simulation analysis results. Specifically, two-levels of crankback are defined—at the intra and inter-domain levels—and active crankback history (failure state) is also leveraged. Furthermore, the proposed solution addresses realistic scenarios where individual domains have full internal visibility via link-state routing, e.g., open-shortest-path-first (OSPF-TE), but generally limited "next-hop" inter-domain visibility, e.g., as provided by inter-area or inter-autonomous system (AS) routing protocols such as hierarchical OSPF or BGP. Although the focus here is on bandwidth provisioning IP/MPLS networks, future adaptations can readily be done for bandwidth-delay settings and even optical wavelength networks.

The paper is organized as follows. Section 2 first presents a survey of the latest work on multi-domain TE provisioning, including standards and research-based activities. Subsequently, Section 3 details the proposed enhanced intra/inter-domain crankback signaling solution. Detailed performance analysis is then conducted in Section 4 and the results compared versus those from counterpart hierarchical inter-domain routing schemes. Finally, conclusions and future research directions are highlighted in Section 5.

2. Background

The IETF has defined a range of TE capabilities for multi-domain provisioning. Foremost, the PCE framework has been introduced to decouple path computation from signaling by defining domain level computational entities. These entities can either reside in a standalone manner or be co-located with nodes and have access to the internal domain resource/policy databases. At the inter-domain level, these PCE entities can interact in a distributed manner to resolve end-to-end routes and two approaches have been defined for varying levels of "global" state, i.e., *per-domain* (PD) and *PCE-based* [3,15]. The former compute TE paths in a "domain-to-domain" manner and are most germane for limited inter-domain visibility. Meanwhile the latter rely on the head-end PCE to computes a *partial* or *loose* route to the destination (domain sequence) and are more suited for increased inter-domain visibility. However, even after path computation, blocking can occur during signaling along a chosen route. Hence new RSVP-TE crankback extensions have also been defined to re-try alternate routes [4]. Specifically, various types of multi-domain crankback frameworks have been outlined (local, intermediate, source-based), but detailed algorithms have not been presented in [4].

Researchers have also studied various multi-domain TE schemes, broadly grouped as *hierarchical routing* or *per-domain* strategies. In the latter, local domain topology/resource state is condensed to generate an "abstracted" graph with fewer vertices and links, e.g., at a designated controller in each domain. This state is then flooded to other domains using hierarchical *link-state* routing between border gateways to build a "global" aggregated graph. As such, these types of schemes can best be classified as PCE-based strategies as per [5]. For example, earlier work in peer group summarization for *asynchronous transfer mode* (ATM) networks has shown very high levels of state reduction [1]. Subsequent studies on multi-domain IP/MPLS networks have also proposed a variety of graph abstractions (e.g., star, mesh, tree, spanner graphs, etc.) to compress bandwidth [6], bandwidth-delay [7,8], and even diversity/survivability [9] information. When coupled with various computation heuristics (such as widest-shortest, shortest-widest, generalized costs, etc.) these schemes yield very good blocking reduction and lower setup signaling overheads.

More recently, topology abstraction/hierarchical routing has also been studied in multi-domain DWDM networks, i.e., to summarize wavelength/converter/risk-group information. For example [10] outlines simple node abstraction for all-optical domains. Meanwhile [11,12] develop full-mesh and star schemes for more realistic multi-domain settings with partial (boundary) conversion. Distributed *routing and wavelength assignment* (RWA) algorithms are also defined to leverage this "global" state. Findings show good inter-domain blocking reduction with full-mesh abstraction (about 20–40% lower than single node abstraction), albeit inter-domain routing overheads are much higher, almost 200–300% higher. Further abstractions for multi-domain optical survivability are also presented in [13].

However, topology abstraction entails significant link-state routing overheads at the inter-domain level, e.g., second level of *open-shortest-path-first* (OSPF-TE) [1,12]. Hence the adoption of this approach may be limited in real-world settings where more scalable distance/path-vector protocols are already well-entrenched. Along these lines, a handful of studies have proposed *signaling-based* crankback strategies for "per-domain" path computation (akin to PCE classification [5]). The goal here is to have individual domains compute their own traversing segments to build a concatenated end-to-end path. For example, [14] defines a basic *"per-domain"* (PD) crankback scheme which probes egress domain nodes for traversal routes and upon failure, notifies upstream border nodes. Overall results show higher request blocking rates and crankback delays, particularly when compared to PCE-based strategies utilizing pre-determined inter-domain routes. Meanwhile, [15,16] detail a modified *compute while switching* (CWS) scheme for MPLS networks. First, a similar crankback procedure to [14] is used to compute an initial inter-domain route, i.e., by probing egress nodes specified by interior and/or exterior gateway protocols. If this search is successful, transmission is started and *simultaneous* crankback is initiated to search for a shorter route, e.g., since random per-domain computation generally does not yield the shortest route. If a shorter route is found, data switchover is performed. Results here show good setup success rates as the scheme essentially mimics an exhaustive-search. However, the CWS scheme entails very high signaling overheads/delays (not analyzed) and requires non-standard extensions to RSVP-TE attributes. Moreover, hitless post-setup flow switchovers may be difficult, especially in GMPLS settings. Finally, [17] addresses end-to-end path delays in multi-domain settings and presents two next-hop domain selection strategies. The first selects the next-hop as the "nearest" egress border node in the domain whereas the other uses tailored inter-domain *round-trip time* (RTT) measurements, i.e., pre-computed global state. Overall the latter heuristic is shown to yield slightly higher carried load and less crankbacks, although it requires adoption of a specialized coordinates system [17].

Overall, these above crankback solutions embody some good initial contributions. However, added innovations are possible for multi-domain settings, e.g., such as novel schemes to limit crankback overheads/delays, improved use of intra/inter-domain crankback (failure) history, and application of available inter-area/inter-AS routing state. This is now addressed further.

3. Enhanced crankback solution

An enhanced multi-domain crankback solution is now presented based upon standard IETF protocols. The solution was recently tabled in [18] and this effort details a more expanded discussion thereof. The framework assumes realistic settings with full link-state routing at the intra-domain level (e.g., OSPF-TE)

and more scalable path/distance vector routing at the inter-domain level. Furthermore, each domain is assumed to have at least one PCE entity with full access to interior and exterior routing databases. This entity plays a key role in the crankback process as it helps resolve next-hop domains (egress border gateways). Meanwhile, all setup signaling is done using the recent crankback framework defined for RSVP-TE [4]. Overall three key innovations are introduced in this work to enhance multi-domain crankback operation, i.e., (1) dual intra/inter-domain crankback counters to limit signaling complexity/delay, (2) full crankback history tracking to improve the re-try process, and (3) intelligent per-domain selection. Details are now presented.

3.1. Multi-domain crankback operation

Before detailing the scheme, the requisite notation is introduced. Consider a multi-domain network comprising of D domains, with the ith domain having n^i nodes and b^i border/gateway nodes, $1 \leqslant i \leqslant D$. This network is modeled as a set of domain sub-graphs, $\boldsymbol{G}^i(\boldsymbol{V}^i,\boldsymbol{L}^i)$, $1 \leqslant i \leqslant D$, where $\boldsymbol{V}^i = \{v_1^i, v_2^i, \ldots\}$ is the set of domain nodes and $\boldsymbol{L}^i = \left\{l_{jk}^{ii}\right\}$ is the set of *intra-domain* links in domain i ($1 \leqslant i \leqslant D$, $1 \leqslant j, k \leqslant n^i$), i.e., l_{jk}^{ii} is the link from v_j^i to v_k^i with available capacity c_{jk}^{ii}. A physical inter-domain link connecting border node v_k^i in domain i with border node v_m^j in domain j is further denoted as l_{km}^{ij} and has available capacity c_{km}^{ij}, $1 \leqslant i, j \leqslant D$, $1 \leqslant k \leqslant b^i$, $1 \leqslant m \leqslant b^j$. Also, $\boldsymbol{B}^i$ denotes the set of border nodes in domain i. Now consider the relevant RSVP-TE message fields. The path route is given by a node vector, $\underline{\boldsymbol{R}}$. Meanwhile, other fields are also defined for crankback as per [4], and include an exclude link vector, $\underline{\boldsymbol{X}}$, to track crankback failure history as well as dual intra/inter-domain crankback counters, h_1 and h_2 (usage will be detailed shortly). Note that [4] only defines a single counter field but bit masking can be used to generate two "sub-counters".

An overview of per-domain computation is first given for the case of *non-crankback* operation, i.e., no resource request failures. Consider a source node fielding a request for x units of bandwidth to a destination node in another domain. This source queries its PCE to determine an egress link to the next-hop domain, e.g., using the *PCE-to-PCE protocol* [3,5]. The PCE then determines the next-hop domain to the destination domain (detailed in Section 3.2) and returns a domain egress border node/link to this domain. Note that this information also contains the *ingress* border node in the downstream domain. Upon receiving the PCE response, the source uses its local OSPF-TE database to compute an *explicit route* (ER) to the specified egress border node. This step searches the k-shortest path sequences over the *intra-domain* feasible links (i.e., $c_{jk}^{ii} \geqslant x$) and chooses the one with the lowest "load-balancing" *cost*, i.e., individual link costs inversely-proportional to free link capacity, i.e., $1/c_{km}^{ij}$. This method is used as it generally outperforms basic hop-count routing, see [6,11]. Granted that an ER path is found above, it is inserted in the path route vector, $\underline{\boldsymbol{R}}$, and RSVP-TE *PATH* messaging is then initiated (along the expanded route) to the ingress border node in the next-hop domain. Here, each intermediate node checks for available bandwidth resources on its outbound link and pending availability, propagates the message downstream. The above procedure is repeated at all next-hop domain border nodes until the destination domain. When the *PATH* message finally arrives at the destination domain, the border node (or PCE) expands the ER to the destination. Upon receiving a fully-expanded *PATH* message, the destination initiates upstream reservation, i.e., by sending a *RESV* message.

Now consider the case of *PATH* processing failure, i.e., due to insufficient bandwidth resources along a route link. Current extensions to RSVP-TE signaling [4] have outlined various alternatives for crankback operation, and two types are chosen for implementation herein, i.e., *intra-domain* (local) and *inter-domain* (intermediate). Namely, the enhanced scheme defines dual crankback counters, i.e., h_1 and h_2, to limit the number of re-try attempts at the intra and inter-domain levels, respectively. Specifically, the above counters are initialized to pre-specified limit values (H_1 and H_2, respectively) in the initial *PATH* message and then decremented during crankback to limit excessive searching along longer and less resource-efficient paths. As such, these values effectively bound the number of intra and inter-domain crankback operations to H_1H_2. Furthermore, crankback failure history is also tracked at both the intra/inter-domain levels.

Using the above counters, two key crankback operations are defined, i.e., *notification* and *re-computation*. The former refers to the (upstream) signaling procedures executed upon link resource failure at an intermediate node, whereas the latter refers to the actual re-routing procedure to select a new route. Now in general, resource signaling (*PATH* processing) failures can occur at *three* different types of nodes, i.e., domain ingress border nodes, domain egress border nodes, and interior nodes. However, in the proposed scheme, only the former performs *re-computation* whereas the latter two simply perform crankback *notification*. These steps are now detailed further.

Crankback notification: Upstream notification is done when there is insufficient bandwidth at an intra-domain link (i.e., at an intra-domain node) or an inter-domain link (i.e., at an egress border node) on an already-expanded ER. This overall algorithm here is shown in Fig. 1. Namely, the *PATH* message is terminated and its appropriate fields updated and copied to an upstream *PATH_ERR* message to the domain's ingress border node. Specifically, the intra-domain counter h_1 is decremented and the failed link is noted. Note that if blocking occurs in the source domain, the *PATH_ERR* is sent back to the source node.

Crankback re-computation: Meanwhile, path re-routing is done by ingress border nodes receiving a *PATH_ERR*. Note that for special case of a source domain (i.e., non-ingress border node), the receiving source node relays the *PATH_ERR* to its PCE for processing. The overall algorithm here is summarized in Fig. 2. Here, two types of crankback re-computations can be done. First consider "intra-domain" crankback. Here, if the intra-domain h_1 counter has not expired in the received *PATH_ERR* message, another next-hop domain/egress border node is selected by the ingress border node (or PCE) for ER expansion. In particular, the exact sequence of next-hop domains tried is pre-computed to try *successively longer* inter-domain routes (i.e., via multi-entry distance vector table, detailed in Section 3.2). Now the enhanced scheme makes full use of crankback history to avoid any failed intra/inter-domain links. Foremost, all failed inter-domain links in $\underline{\boldsymbol{X}}$ that egress from the domain are removed from consideration, i.e., only consider "non-failed" next-hop domain egress links. Additionally, all intra-domain links listed in the exclude link vector $\underline{\boldsymbol{X}}$ are also precluded from *local* ER computation. Note that the route vector $\underline{\boldsymbol{R}}$ is also searched to make sure that an upstream domain is not traversed twice, i.e., no "domain level" loops. Regardless, it still may

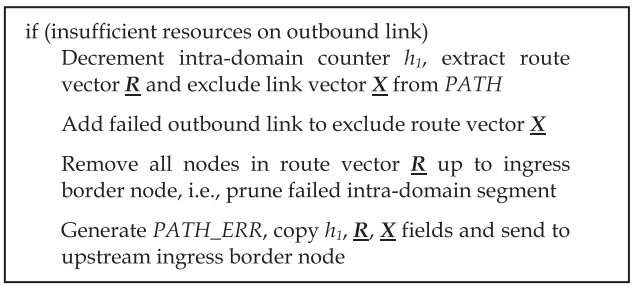

Fig. 1. Crankback notification algorithm (at local or egress border node).

```
/* Attempt intra-domain re-routing */
if (h1 not expired)
    Select next-hop domain/egress link using multi-entry
        distance vector table s.t. next-hop domain is not in R
        and egress link is not in X

    if (next hop egress node found)
        Make copy of local network graph (via IGP database),
        prune all local failed links listed in X, compute new ER
        to egress border node

        if (LR expansion successful)
            Initiate PATH signaling to new egress node
            intra_domain_crankback_done=1;

/* Attempt inter-domain re-routing */
if (!intra_domain_crankback_done & h2 not expired)
    Decrement inter-domain counter h2, extract route vector R
    and exclude route vector X from PATH

    Add ingress inter-domain link to exclude link vector X

    Remove all nodes in route vector R up to previous
    domain's ingress border node

    Copy h2, R, X fields, reset h1=H1, generate PATH_ERR and
    send to previous domain's ingress border node
else
    Copy h1, h2, R, X fields,generate PATH_ERR,send to source
```

Fig. 2. Crankback re-computation algorithm (at domain ingress border node).

not be possible to initiate/establish a domain-traversing route for various reasons, i.e., h_1 counter expired, LR expansion failure to selected egress node, or all egress border links in exclude link vector $\underline{\boldsymbol{X}}$, etc. In these cases, the ingress border node must initiate a more globalized "inter-domain crankback" response via a *PATH_ERR* message to the ingress node in the *upstream* domain in the *PATH* route vector $\underline{\boldsymbol{R}}$ (or source node if upstream domain is source domain). To improve history tracking, the ingress border node also inserts its own *ingress link* in the exclude route vector of the *PATH_ERR*, i.e., to avoid future re-tries on this link. Note that "inter-domain crankback" is only initiated if the inter-domain crankback counter, h_2, is non-zero, otherwise the request is failed (i.e., *PATH_ERR* to source, Fig. 2).

An example of crankback notification is shown in Fig. 3 for interior and egress border nodes ($H_1, H_2 = 2$). For example, consider bandwidth blocking on the link l_{42}^{ii}, i.e., step 1. Here, the interior node v_4^i prunes the route vector $\underline{\boldsymbol{R}}$ to the domain ingress node, adds the blocked link to the exclude route vector $\underline{\boldsymbol{X}}$, decrements the intra-domain counter h_1, and sends all this information back to the ingress node v_1^i via a *PATH_ERR* message. A similar procedure is also shown for blocking at the egress border node v_3^i (i.e., step 2, Fig. 3). Sample crankback re-computation is also shown in Fig. 1. For example when blocking initially occurs on link l_{42}^{ii}, the ingress border node v_1^i re-tries intra-domain path expansion to egress border node v_3^i. When this second intra-domain attempt fails at the egress link l_{31}^{ii+2}, ingress node v_1^i receives a *PATH_ERR* with a zero h_1 counter. In response, it marks its ingress link l_{21}^{i} as failed, prunes the route to the ingress border node in previous domain $i-1$, i.e., node v_1^{i-1}, and sends a *PATH_ERR* message (step 3, Fig. 3). The upstream ingress border node v_1^{i-1} decrements h_2, resets the h_1 counter to H_1, and then initiates a re-try to a new egress border node, v_3^{i-1} (step 4, Fig. 1). Note that if the previous domain is the source domain, the *PATH_ERR* is simply sent to the source.

Overall, the proposed crankback operation relies upon a series of static table lookups and intra-domain path computation (loose route expansion). Here the individual path computations run the Dijkstra shortest-path algorithm, which is of $O(|\boldsymbol{V}_i|log|\boldsymbol{V}_i|)$ complexity for domain i. Hence assuming a maximum domain size of $N > |\boldsymbol{V}_i|$, the resultant path computation complexity here is $O(H_1H_2N\log N)$.

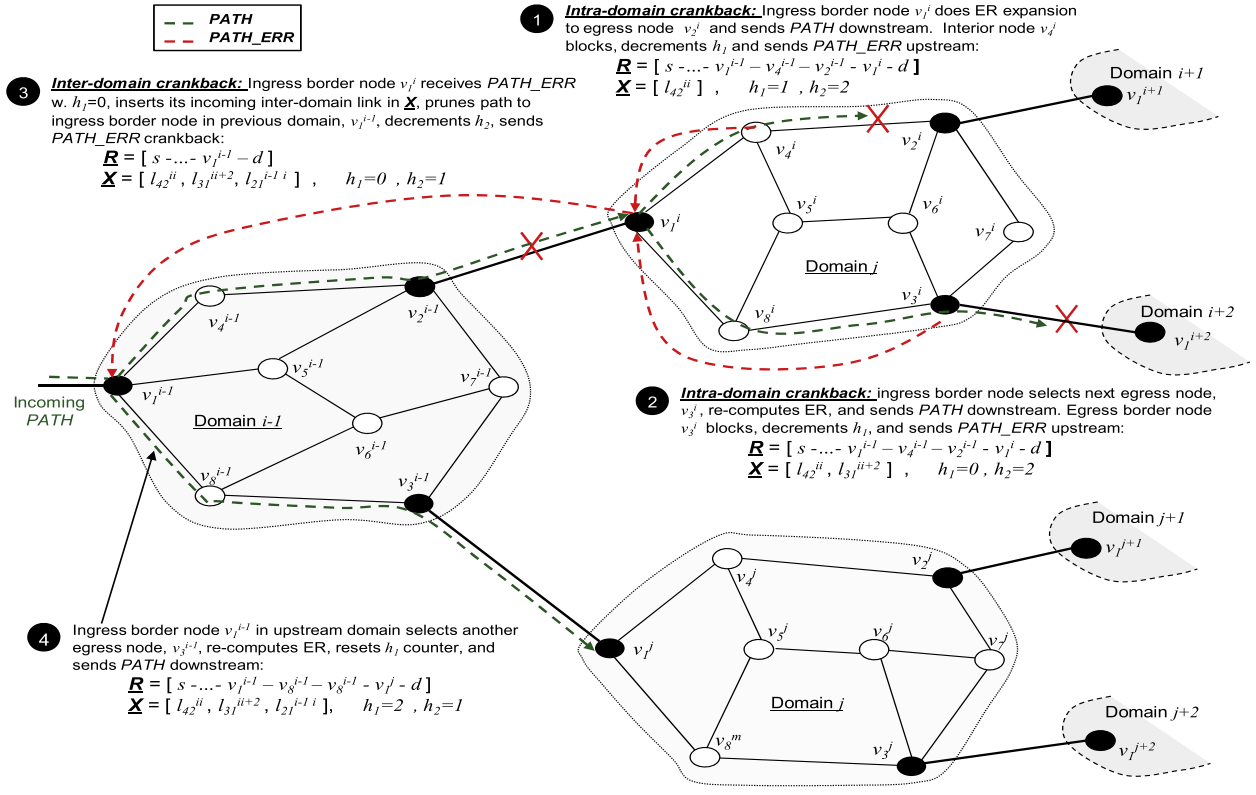

Fig. 3. Enhanced intra/inter-domain crankback scheme ($H_1 = 2$, $H_2 = 2$).

3.2. Next-hop domain computation

As mentioned earlier, a key provision in the enhanced crankback scheme is the use of existing inter-domain state to improve the search process. This is achieved by pre-computing a *multi-entry* distance vector table at all domain border nodes (or PCE) to list up to K next-hop domains/egress links to each destination domain. Namely, at domain i, the kth table entry to a destination domain j, $T^i(j,k)$, is computed as the egress inter-domain link (to the next-hop domain) on the kth shortest "domain level" hop-count path to domain j ($1 \leqslant i, j \leqslant D$, $i \neq j$, $1 \leqslant k \leqslant K$). Clearly the number of entries to a destination will be upper-bounded by the minimum of K and the maximum number of inter-domain links that egress from the domain.

Now consider the actual computation of this table at a border node (or PCE) in domain i, the algorithm for which is summarized in Fig. 4. Here a "simple node" [2] view of the global topology is firstderived, i.e., $\boldsymbol{H}(\boldsymbol{U},\boldsymbol{E})$, where $\boldsymbol{U}$ is the set of domains $\{\boldsymbol{G}^i\}$ reduced to vertices and $\boldsymbol{E}$ is the set of inter-domain links $\left\{l_{km}^{ij}\right\}, i \neq j$. At the inter-area level, this graph can be obtained from hierarchical *open-shortest-path-first* (OSPF) link-state databases whereas at the inter-AS level it can approximately be deduced from *border gateway protocol* (BGP) path vector state (albeit not all inter-domain connectivity may be visible due to policy restrictions). An iterative shortest-path scheme is then used to compute multiple routes to all destination domains over $\boldsymbol{H}(\boldsymbol{U},\boldsymbol{E})$. Namely, the scheme basically loops over all destination domains $j \neq i$ (index j) and computes up to K next-hop egress links (index k) over a temporary copy of $\boldsymbol{H}(\boldsymbol{U},\boldsymbol{E})$, i.e., $\boldsymbol{H}'(\boldsymbol{U},\boldsymbol{E})$. At the kth iteration, the scheme computes the shortest "domain level" hop-count path to the destination domain using $\boldsymbol{H}'(\boldsymbol{U},\boldsymbol{E})$, and if found, stores the egress link from the source domain in $T^i(j,k)$. This link is then pruned from $\boldsymbol{H}'(\boldsymbol{U},\boldsymbol{E})$ and the procedure repeated to compute the next shortest "domain level" hop-count path. The procedure is terminated if all K entries are filled and/or the vertice for domain i in $\boldsymbol{H}'(\boldsymbol{U},\boldsymbol{E})$ becomes disconnected. Hence the next-hop domain selection procedure during crankback re-computation (as detailed in Section 3.1) simply searches these K table entries, $T^i(j,k)$, to a destination domain j in increasing order. This sequentially drives the crankback search along fixed "domain level" sequences of increasing length, but with provisions to prune "failed" entries (in $\underline{\boldsymbol{X}}$). Overall, these entry tables will be relatively static if inter-domain topology changes are relatively infrequent.

Now as mentioned in the survey in Section 2, there have been some recent studies on "per-domain" multi-domain crankback [14–16]. However, a key differentiating feature of the proposed scheme herein is its (above-detailed) computation and use of multi-hop domain tables. For example, all existing schemes largely select next-hop domains in a random manner [15,16] or based upon minimum intra-domain hop counts [13]. Specifically, in the "per-domain" scheme of [13] the head-end ingress node (fielding a *PATH* message) selects the egress border node with the shortest intra-domain path (hop-count). If subsequent signaling failure occurs, intra-domain crankback is attempted to an egress border node with the next shortest intra-domain path, and so on. It is evident here that this strategy only focuses on intra-domain resource minimization and will clearly not yield the shortest (or otherwise optimal by another metric) path at the *inter-domain* level. By contrast, the proposed algorithm in Fig. 4 pursues much more of a inter-domain cost-minimization strategy, albeit based upon static topological state.

```
Generate simple-node abstraction of global topology via
EGP database information, i.e., H(U, E)
/* At domain i, loop across all possible destination domains */
for j = 1 to D
  if (j ≠ i)
    Make temporary copy of graph H(U,E), i.e., H'(U,E)
    /* Compute up to K table entries */
    for k=1 to K
      Compute shortest-path from domain i to j in H'(U,E)
      if (shortest path route found)
        Save route line from domain i in k-th table entry
        T^i(j,k), i.e., link from domain i vertice in H'(U,E)
        Prune above-selected link from H'(U,E)
      if (domain i becomes disconnected)
        break k-loop
```

Fig. 4. Multi-entry distance vector table computation algorithm (at PCE).

4. Performance evaluation

The performance of the enhanced multi-domain crankback solution is tested by developing specialized models in *OPNET Modeler*™. Tests are done using two multi-domain backbone topologies, including a 10-domain topology with 25 inter-domain links and a modified NSFNET topology (with nodes replaced by domains) with 16 domains/25 inter-domain links, see Figs. 5 and 6. This extends upon the work in [18] which only considers one topology. Here the 10-domain topology has an average of 2.5 links/domain whereas the NSFNET topology has an average of 1.56 links/domain, i.e., slightly lower inter-domain connectivity. Furthermore, the average domain size in each network is set to about 10–12 nodes. Carefully note that multi-homed interconnection is also used in the 10-domain topology to reflect realistic settings, e.g., dual-homing between domains 4 and 7. Furthermore, all link capacities are set to 10 Gbps and connection requests sizes are varied from 200 Mbps to 1 Gbps in increments of 200 Mbps, i.e., to model realistic fractional Ethernet demands. Here, all connections are generated between random nodes in randomly-selected domains and each run is averaged over 2,50,000 connections with mean holding times of 600 s (exponential). Meanwhile, request inter-arrival times are also exponential and varied with load. Finally, a maximum of $K = 5$ next-hop domain entries are computed in the distance vector table, although the number searched is limited by the H_2 value set in the simulation run.

A key objective here is to compare crankback performance against hierarchical inter-domain routing with topology abstraction, i.e., simple node, full-mesh [6,7,11]. Consider the details of the latter scheme. In full-mesh abstraction, the PCE computes "abstract links" to condense trans-domain routes, $O(|\boldsymbol{B}^i|(|\boldsymbol{B}^i|-1))$ state. Here, the capacity of an abstract link is derived as the mean bottleneck capacity of the k-shortest-paths between the respective border nodes [7,11]. These links (along with physical inter-domain links) are then advertised using a second level of OSPF-TE between border nodes [1]. Namely, link updates are generated using *significance change factors* (SCF) and hold-off timers [1], and the respective values are set to 10% (SCF) and 200 s (hold-off timer). This inter-domain link-state is then used to build a "global" topology for computing/expanding end-to-end *loose-routes* (LR). Meanwhile in simple node abstraction, all domains are condensed to virtual nodes, i.e., no domain-internal state advertised, only *physical* inter-domain link-state. Finally, the exhaustive-search *per-domain* (PD) crankback scheme of [14] is also tested here for comparison sake (which does not incorporate failed intra-domain link pruning or intelligent next-hop domain selection).

Crankback performance is first evaluated for the case of inter-domain only connections, i.e., no local intra-domain requests. Here

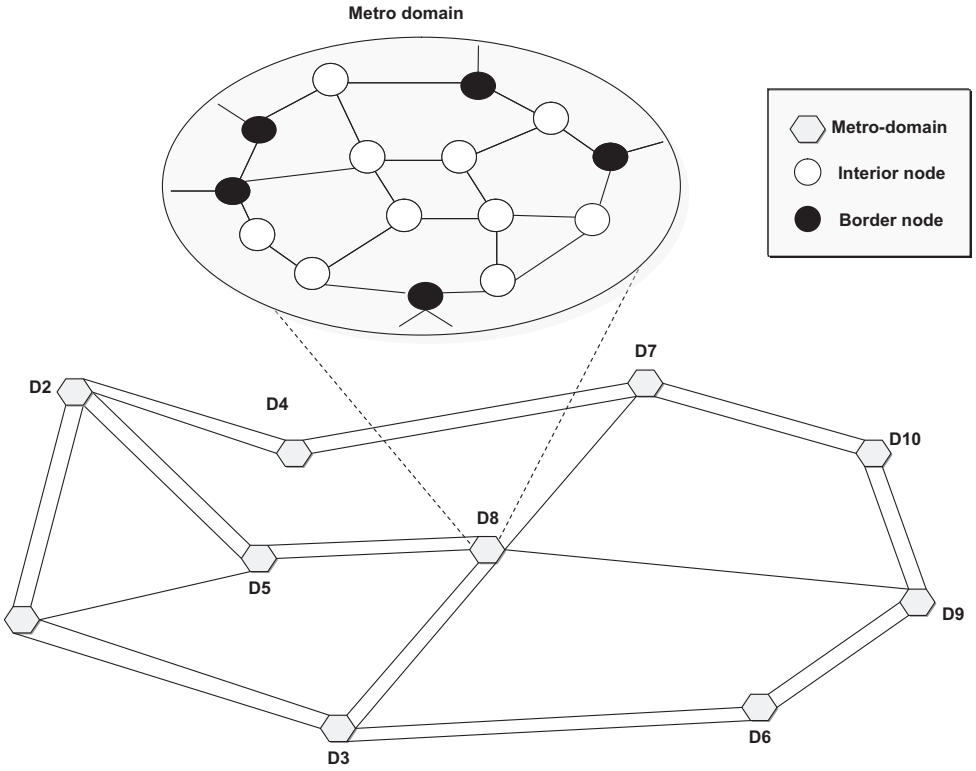

Fig. 5. 10-Domain test topology.

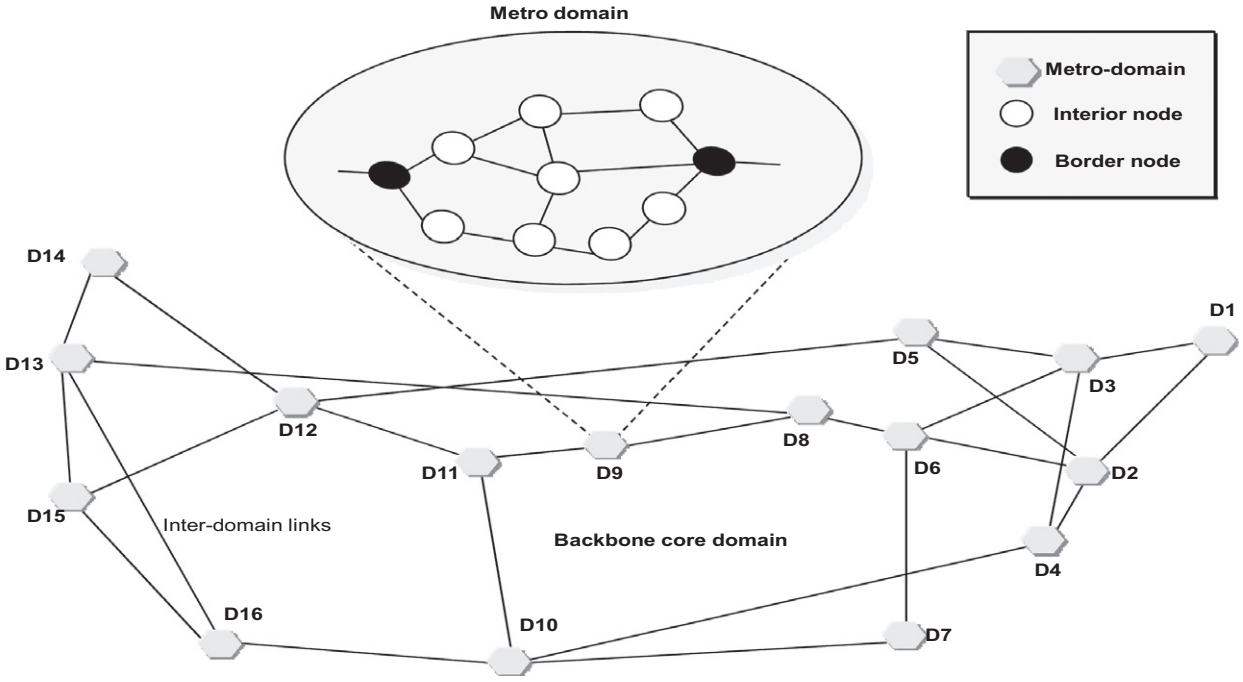

Fig. 6. 16-Domain modified NSFNET test topology.

the inter-domain *bandwidth blocking rates* (BBR) are plotted for the various schemes in Fig. 7 ("HR" denotes hierarchical routing, "CB" denotes crankback, and "PD" denotes the scheme in [14]). Moreover, several configurations are tested for the enhanced crankback scheme, including intra-domain only ($H_1 = 3/H_2 = 0$), inter-domain only ($H_1 = 0/H_2 = 2$), and joint ($H_1 = 2/H_2 = 2$, $H_1 = 3/H_2 = 3$). First of all, the results for both network topologies indicate that the enhanced scheme gives the best performance when both intra and inter-domain crankback is enabled, i.e., inter-domain only crankback with $H_1 = 0$ gives highest blocking. Next, it is also seen that blocking reduction tends to level off after moderate crankback levels, e.g., the blocking performance for $H_1 = 2/H_2 = 2$ closely matches that for $H_1 = 3/H_2 = 3$ and is notably better than that with the more exhaustive PD crankback scheme [14]. In general, this is due to the

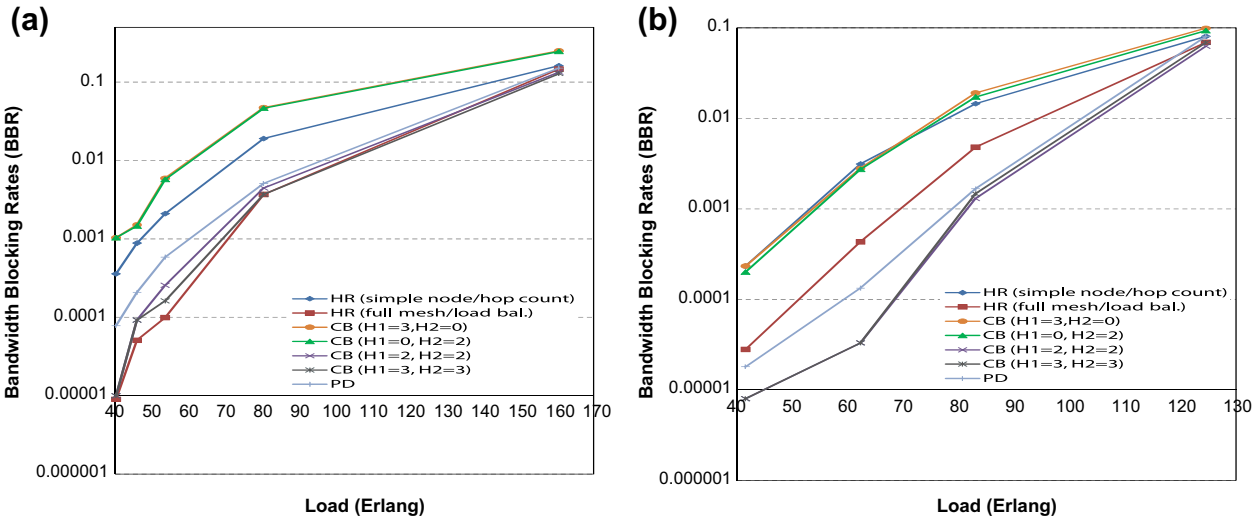

Fig. 7. Inter-domain BBR performance: (a) 10-domain, (b) NSFNET.

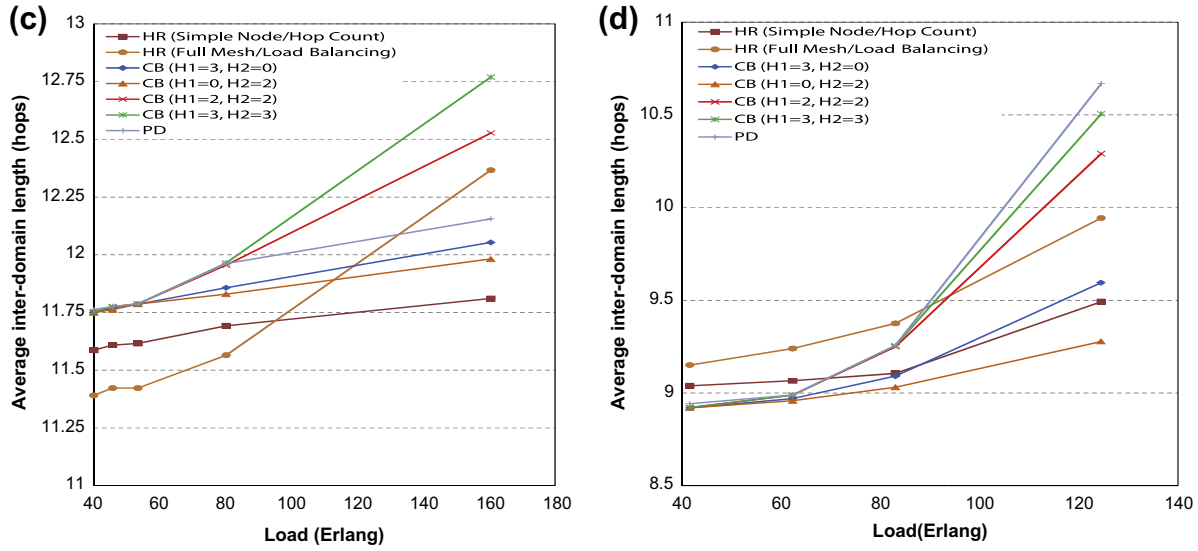

Fig. 8. Average inter-domain path length: (a) 10-domain, (b) NSFNET.

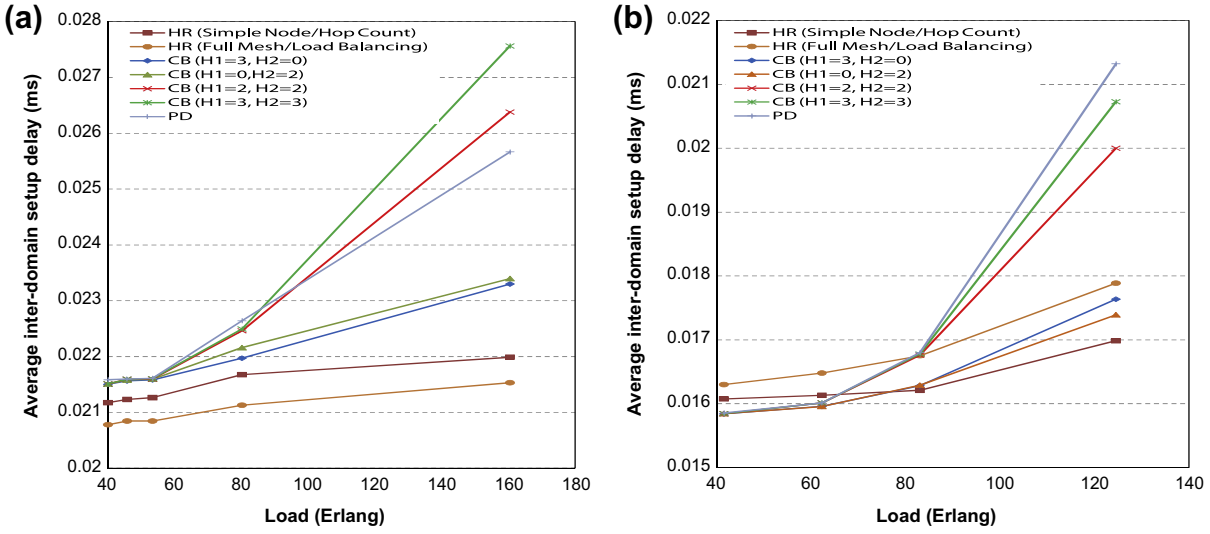

Fig. 9. Average connection setup delay (a) 10-domain, (b) NSFNET.

fact that excessive crankback attempts yield increased route lengths and higher bandwidth fragmentation. Finally, the results in Figs. 7 and 8 show that the proposed crankback solution (with moderate counter values, i.e., $H_1 = 2/H_2 = 2$) can even outperform the other, more complex hierarchical routing strategies, extending upon the findings in [18]. In particular, resultant BBR values are always lower than those yielded by simple node abstraction, and for the case of NSFNET, even lower than those yielded by more advanced full-mesh abstraction. This is a very significant gain, given the fact that associated crankback messaging overheads (not shown here) are over an order magnitude lower than hierarchical routing message loads.

Next, the resource usage/efficiencies of the respective schemes are gauged by plotting the average inter-domain path lengths in Fig. 8. In both of topologies, it is seen that increased *inter-domain* crankback levels (i.e., $H_2 = 2$ or 3, exhaustive PD scheme [14]) result in the highest utilizations, particularly at higher loads. Moreover, these usage levels are also higher than those for the hierarchical routing schemes running simple node and/or full-mesh abstraction. Nevertheless, such increases are generally expected when performing "per-domain" crankback operation, and the results show that the maximum increases are bounded by 10% even at high loads. In addition, end-to-end setup delays for successful connections are also plotted in Fig. 9 for the two topologies, assuming 1.0 ms link delays and 0.05 ms node processing delays. Again, these results show that the proposed crankback scheme generally gives higher setup delays when running both intra and inter-domain crankback, i.e., as compared with hierarchical routing. However, these increases are generally bounded in the 15–20% range and are most pronounced at very high loads (over 10% BBR).

Inter-domain crankback performance is further evaluated in the presence of (non-negligible) interfering *intra-domain* connection loads. Specifically, all domains are seeded with relatively light intra-domain connection arrivals to simulate competing cross-traffic for inter-domain requests, i.e., tuned to give about 1% BBR for intra-

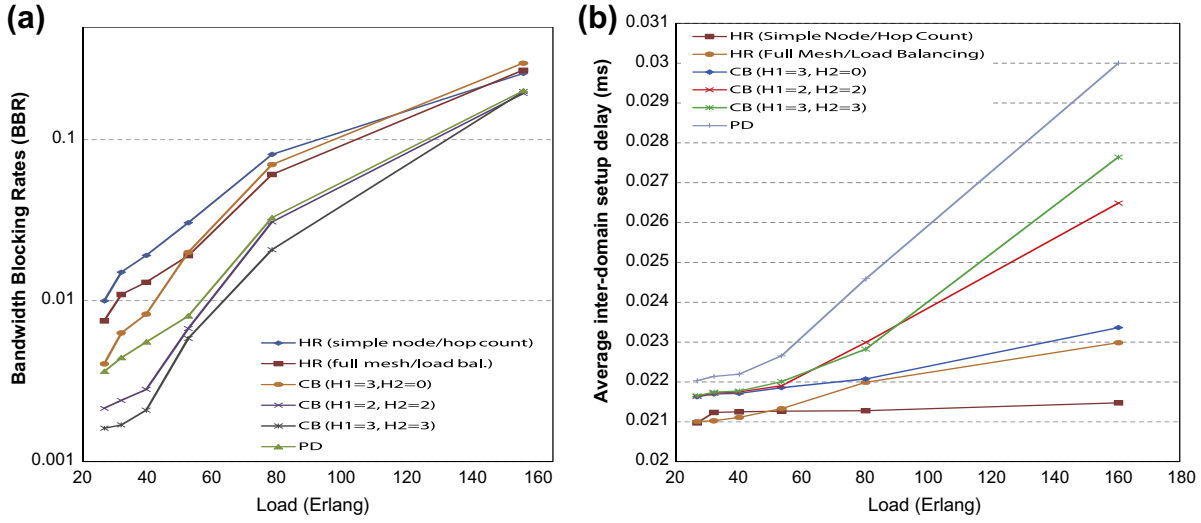

Fig. 10. (a) BBR (crankback, hierarchical routing), (b) Average connection setup delay.

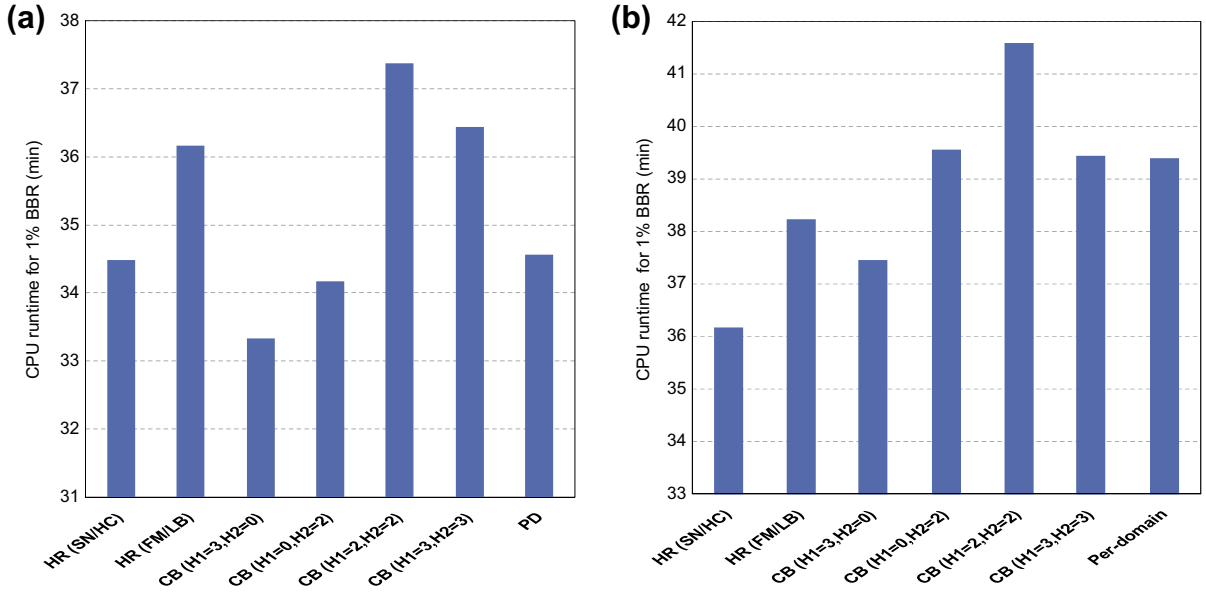

Fig. 11. Simulation run times (a) 10-domain, (b) NSFNET.

domain requests. The resulting BBR and average setup delay results are plotted in Fig. 10(a) and (b), respectively, for the 10-domain topology. Foremost, the findings show that the enhanced crankback scheme gives much lower BBR performance than any of the competing hierarchical routing schemes. In fact, the blocking reduction between the $H_1 = 3/ H_2 = 3$ and full-mesh abstraction schemes is close to an order of magnitude at low-to-medium loads, a notable increase in separation versus the inter-domain only loading case, i.e., compare Fig. 7(a) and Fig. 10(a). Moreover, larger intra/inter-domain counter values, e.g., $H_1 = 3/H_2 = 3$ versus $H_1 = 2/H_2 = 2$, also give better BBR reduction as they are more effective in handling increased intra-domain blocking. As expected, the proposed crankback scheme also gives higher average setup delays as compared to full-mesh hierarchical routing, but the differentials are within 20%. Nevertheless, these average values are still lower than those with the exhaustive PD crankback scheme, by about 5–10%. Finally, similar runs for the NSFNET topology (i.e., with added intra-domain traffic loads, not shown) also confirm much better BBR performances with the proposed crankback scheme.

Finally, the actual simulation run times are also measured to get a gauge on the run-time complexity of the multi-domain provisioning schemes (for 1,00,000 random connection requests at about 1% BBR operating point). These findings are shown in Fig. 11(a) and (b) for the 10-domain and modified NSFNET topologies, respectively. In general, it is seen that the proposed crankback scheme yields higher run times when the crankback counter values are high, and these values are slightly larger than those for the hierarchical routing strategy with full-mesh abstraction. Nevertheless, it is interesting to note that increased levels of crankback can actually reduce run times, e.g., 41.6 min with $H_1 = 2/H_2 = 2$ and 39.4 min with $H_1 = 3/H_2 = 3$ for NSFNET (Fig. 11(b)). These findings indicate that the increased intra-domain counter values (which result in more intra-domain re-tries) actually lead to faster connection setup responses, either success or failure.

5. Conclusions

This paper studies crankback signaling in multi-domain settings and introduces several key innovations. Namely, a dual crankback counter approach is used to limit the number of intra/inter-domain crankback attempts. In addition, crankback history in the form of link failure state is also leveraged at both the intra and inter-domain levels in order to improve the overall success and speed of the setup process. Finally, improved next-hop domain selection strategies are developed to drive the overall search process by using existing (limited) inter-domain routing state. Detailed performance results show much-improved blocking performance with the proposed scheme, i.e., as compared with complex hierarchical inter-domain routing strategies (with topology abstraction) as well more exhaustive "end-to-end" crankback schemes. These gains are particularly notable for the case of non-negligible intra-domain connection loads. Future studies will look at extending this work for post-fault restoration survivability.

Acknowledgements

This research has been supported in part by the Department of Energy Office of Science under Award# ER25828 and the National Science Foundation (NSF) under Award# CNS-0806637. The authors are very grateful to these agencies for their support.

References

[1] N. Ghani et al., Control plane design in multidomain/multilayer optical networks, IEEE Communications Magazine 46 (6) (2008) 78–87.
[2] R. Zhang, J. Vasseur, MPLS inter-autonomous systems traffic engineering (TE) requirements, IEFT RFC 4226, November 2005.
[3] J. Ash, J. Le Roux, A path computation element (PCE) communication protocol generic requirements, IETF RFC 4657, September 2006.
[4] A. Farrel et al., Crankback signaling extensions for MPLS and GMPLS RSVP-TE, IETF Request RFC 4920, July 2007.
[5] P. Torab et al., On cooperative inter-domain path computation, IEEE ISCC 2006, Sardinia, Italy, June 2006.
[6] F. Hao, E. Zegura, On scalable QoS routing: performance evaluation of topology aggregation, IEEE INFOCOM 2000.
[7] T. Kormaz, M. Krunz, Source-oriented topology aggregation with multiple qos parameters in hierarchical networks, ACM TOMACS 10 (4) (2000) 295–325.
[8] K. Liu et al., Routing with topology abstraction in delay-bandwidth sensitive networks, IEEE/ACM Transactions on Networking 12 (1) (2004) 17–29.
[9] A. Sprintson et al., Reliable routing with QoS guarantees for multi-domain IP/MPLS networks, IEEE INFOCOM 2007, Alaska, May 2007.
[10] S. Sanchez-Lopez et al., A hierarchical routing approach for GMPLS-based control plane for ASON, IEEE ICC 2005, Korea, June 2005.
[11] Q. Liu et al., Hierarchical routing in multi-domain optical networks, Computer Communications 30 (1) (2006).
[12] Q. Liu, C. Xie, T. Frangieh, N. Ghani, A. Gumaste, N. Rao, T. Lehman, Inter-domain routing scalability in optical DWDM networks, IEEE ICCCN 2008, US Virgin Islands, August 2008.
[13] D. Truong, B. Thiongane, Dynamic routing for shared path protection in multi-domain optical mesh networks, OSA Journal of Optical Networking 5 (1) (2006) 58–74.
[14] S. Dasgupta, J.C. de Oliveira, J.P. Vasseur, Path-computation-element-based architecture for interdomain MPLS/GMPLS traffic engineering: overview and performance, IEEE Network 21 (4) (2007) 38–45.
[15] F. Aslam et al., Interdomain path computation: challenges and solutions for label switched networks, IEEE Communications Magazine 45 (10) (2007) 94–101.
[16] F. Aslam et al., Inter-domain path computation using improved Crankb+ack signaling in label switched networks, IEEE ICC 2007, Glasgow, Scotland, June 2007.
[17] C. Pelssner, O. Bonaventure, Path selection techniques to establish constrained interdomain MPLS LSPs, in: Proceedings of IFIP International Networking Conference, Coimbra, Portugal, May 2006.
[18] F. Xu, et al., Enhanced crankback signaling for multi-domain traffic engineering, IEEE ICC 2010, Cape Town, South Africa, May 2010.