MPLS-TE Routing: Adopting a Generic Architecture and Evaluating Various Implementation Approaches

Imène Chaieb, Jean-Louis Le Roux, Bernard Cousin

To cite this version:

Imène Chaieb, Jean-Louis Le Roux, Bernard Cousin. MPLS-TE Routing: Adopting a Generic Architecture and Evaluating Various Implementation Approaches. Australian Telecommunication Networks and Applications Conference (ATNAC 2006), Dec 2007, Melbourne, Australia. pp.100-105. hal-00180771

HAL Id: hal-00180771
https://hal.science/hal-00180771

MPLS-TE Routing: Adopting a Generic Architecture and Evaluating Various Implementation Approaches

Imene Chaieb and Jean-Louis Le Roux
France Telecom R&D, Lannion 22307, France
Email: {imene.chaieb, jeanlouis.leroux}@francetelecom.com

Bernard Cousin
IRISA, Rennes 35042, France
Email: bernard.cousin@irisa.fr

Abstract— **Various Multi-Protocol Label Switching-Traffic Engineering (MPLS-TE) Routing Systems have been proposed in the literature to achieve optimization of resources utilization, Quality-of-Service (QoS) and Fast Recovery. This paper proposes a generic architecture for MPLS-TE Routing Systems, which aims to ease the classification and the analysis of these systems. Then this paper defines a set of MPLS-TE classification criteria. The combination of these criteria leads to the identification of main families of MPLS-TE Routing Systems which are finally compared and qualitatively evaluated according to a set of metrics.**

I. INTRODUCTION

The emergence of multi-service IP-centric networks, which transport value added services such as VoIP (Voice/Video Telephony over IP), IP TV, Video on Demand and VPN traffic, leads to the requirements for strict QoS delivery (delay, jitter, packet loss), and high availability. Given the drastic increase of last miles capacities, and the reduction of the gap between core and access bandwidth, the over-provisioning approaches followed for years by operators in core and backhaul networks, so as to ensure QoS, are no longer a panacea today in backhaul networks, and may no longer be a valuable approach in core networks at mid term. Hence Traffic Engineering (TE) mechanisms are required so as to optimize network resource utilization, that is to maximize the amount of traffic that can be transported while ensuring the quality of service, with as main objective to reduce network costs and postpone investments. In order to address the traffic increase and satisfy the QoS requirements of multimedia applications, various TE mechanisms are proposed, among those MPLS-TE, a connection oriented mechanism based on the MPLS forwarding paradigm, well suited to TE thanks to its Explicit Routing capabilities. The MPLS-Traffic Engineering approach [1] allows setting up explicitly routed Traffic Engineering-Label Switched Path (TE-LSP) whose path satisfy a set of traffic engineering constraints, including bandwidth. MPLS-TE combines explicit routing capabilities of MPLS with a constraint based routing paradigm based on dynamic resources discovery (ISIS-TE [2], OSPF-TE [3]), constrained path computation, and distributed LSP signalling and resources reservation (RSVP-TE) [4]. MPLS-TE ensures Traffic Engineering functions such as network resources optimization, strict QoS guarantees, and fast recovery upon link or node failures. For a load balancing purpose, a set of two or more TE-LSPs may be used to route a given aggregate traffic demand between two end points. The TE-Trunk concept defined in [1] allows accounting for such load balancing. A TE-Trunk is defined as a set of one or more LSPs used to carry an aggregate traffic demand between two points for a given service class. A TE-Trunk is characterized by its reserved bandwidth and a set of TE parameters (e.g. class of service, delay...). In order to efficiently route flows in TE-Tunks, complementary mechanisms are required on top of the standard MPLS-TE control plane. This includes essentially a TE-Trunk Utilization function, responsible for an efficient routing of a set of N flows in a set of M TE-LSPs of one or more TE-Trunks, along with an Adaptability mechanism responsible for adapting the TE-Trunks (LSPs resizing/creation/suppression) according to traffic matrix changes and/or topology modifications (failures). These Utilization and Adaptability functions are actually intimately linked to the MPLS-TE Path Computation function. The combination of the MPLS-TE control plane building blocks (Routing, Path Computation, Signaling) with these additional functions (Utilization and Adaptability) form together what we call a MPLS-TE Routing System. In the literature, there are papers that focus on MPLS-TE Path Computation. Some of these solutions ([5] [6] [7]) propose efficient algorithms to place TE-LSPs in networks and satisfy a pre-defined set of flow requests, others take interest to this functionality but in case of network failure. There are also papers which account for TE-LSPs Utilization [8] and for Adaptation mechanisms [9]. Others, study flow admission control and its application in MPLS-TE networks [10]. However, a global study that covers the overall architecture of an MPLS-TE Routing System is not considered. In the remainder of this paper, we firstly propose, a generic architecture to describe the functions of a MPLS-TE Routing System and their interaction. Then, we rely on this architecture to define a set of MPLS-TE classification criteria and we combine these criteria, so as to identify main MPLS-TE System families. Finally we propose a qualitative evaluation and comparison of these families according to a set of evaluation metrics.

II. FUNCTIONAL ARCHITECTURE OF MPLS-TE ROUTING SYSTEMS

In this section we propose a generic architecture to describe an MPLS-TE Routing System (Fig. 1). This is a functional architecture that helps in covering a large solution spectrum. It is comprised of a set of functions also called building blocks. Some of these building blocks are running on routers, others may be running either on routers or on one or more network servers. We distinguish standard MPLS-TE blocks and implementation specific blocks:

- Standard MPLS-TE functions include the TE Topology Discovery function ensured by an IGP-TE protocol (either OSPF-TE or ISIS-TE) and the LSP Signalling function ensured by the RSVP-TE protocol. These standard functions are located in routers.
- Implementation specific functions include the TE-Trunk Agent, the TE-Trunk Path Computation, the TE-Trunk Adaptation and the TE-Trunk Utilization functions. These functions may be located in routers or externalized in one or more network servers. This also comprises the TE-Manager function which is always located in a network server.

In the below sections we focus on the five building blocks in charge of resource optimization: TE-Manager (TM), TE-Trunk AGent (TAG), TE-Trunk Computation (TC), TE-Trunk ADaptation (TAD), and TE-Trunk Utilization (TU).

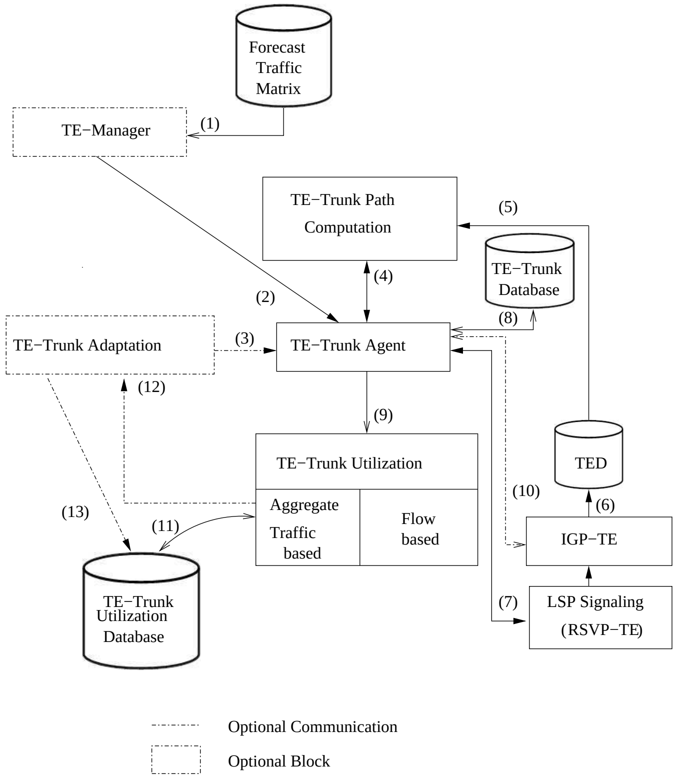

Fig. 1. Generic architecture for MPLS-TE Routing Systems

A. TE-Manager

The TE-Manager (TM) is a functional entity that takes the decision to setup/release/modify TE-Trunks by relying on the forecast traffic matrix (1), that is the set of aggregate traffic demands between each pair of Edge Routers. It sends TE-Trunk setup/deletion/modification requests to the set of one or more TE-Trunk Agents (2). This function is optional, TE-Trunks may be defined by the operator and may be directly configured on TE-Trunk Agents.

B. TE-Trunk Agent

The TE-Trunk Agent (TAG) is the heart of the architecture. It controls the TE-Trunks establishments/modifications/deletions in the network. It coordinates the actions of the TE-Manager, the TE-Trunk Adaptation, the TE-Trunk Path computation, the TE-Trunk Utilization and the LSP Signalling blocks. It handles TE-Trunk setup/deletion/modification requests sent by the TE-Manager (1), and TE-Trunk modification requests sent by the TE-Trunk Adaptation block (3). It sends TE-Trunk Computation requests to the TE-Trunk Path Computation block (4). Once paths are computed the TE-Trunk Agent sends LSP setup requests to the RSVP-TE module (7) so as to signal the TE-LSPs along the computed paths. Once the TE-LSPs are setup, the TE-Trunk Agent feeds the TE-Trunks Database which contains information related to the established TE-Trunks (TE-Trunk constraints, TE Trunk paths, etc.) (8). It also communicates the established TE-Trunks and the corresponding LSPs to the TE-Trunk Utilization block (9). In an "Online mode", it may communicate with the IGP-TE (10) and LSP Signalling blocks so as to be notified of network (link/node) and TE-LSPs failures. This communication allows the TE-Trunk Agent to detect failures and call the TE-Trunk Path Computation block so as to reroute the TE-Trunks on paths avoiding failed elements.

C. TE-Trunk Path Computation

The TE-Trunk Computation block (TC) is a fundamental building block in MPLS-TE Routing Systems. It has to find TE-Trunks paths by operating on the Traffic Engineering Database (TED) (5) fed up by the IGP (6) and considering the TE-Trunks constraints. It handles Trunk Computation requests sent by the TE-Trunk Agent. A request may correspond to a single TE-Trunk or to a set of TE-Trunks. The request may be a Trunk setup request or a Trunk modification one. The output for a given Trunk is a path or a set of paths whose cumulative bandwidth fits the Trunk(s) request.

D. TE-Trunk Adaptation

The TE-Trunk Adaptation block (TAD) is in charge of adapting TE-Trunk size to the actual traffic load. It increases TE Trunk size (i.e. it increases the amount of bandwidth reserved for the TE-Trunk), so as to anticipate congestion issues, when the load between a pair of nodes increases; and decreases TE Trunk size so as not to waste unused bandwidth when the load between a pair of nodes decreases. Verification of the TE-Trunk load can be done in a *timer driven* manner, in which case the TE-Trunk load in the TE-Trunk Utilization databases is periodically checked by the Adaptation block (13) or it can also be done in an *event driven* manner, in which case the TE-Trunk Utilization block notifies the Adaptation block that a TE-Trunk is congested or is going to be congested (12). Note that this block is optional and may not be used in every MPLS-TE Routing Systems.

III. IMPLEMENTATION CONSIDERATION: DISCUSSION AND EVALUATION

In the previous section we proposed a functional architecture for MPLS-TE systems, which includes in addition to standard MPLS-TE blocks, specific blocks such as TE-Trunk Computation, TE-Trunk Adaptation and TE-Trunk Utilization. This architecture may help classifying MPLS-TE mechanisms and improve the design of MPLS-TE systems. An MPLS-TE routing system corresponds actually to a specific implementation of this architecture. The blocks of this generic architecture may be located in different elements (Centralized on Network servers or distributed in Edge routers). The performances of an MPLS-TE routing system, in terms of scalability, reactivity and optimality actually depend on various implementation options, including the repartition of the functions. Before discussing these options, a description of some classification criteria which will help the discussion, is proposed.

A. MPLS-TE classification criteria

Several criteria are identified to arrange the various approaches for implementing an MPLS-TE Routing Systems. We distinguish the following:

1) Time Scale:
    - Offline (Off): TE-Trunks are computed and established periodically based on forecast traffic matrices. This mode allows more time for path computation. This implies that there is no TE-Trunk Adaptation and there is no LSP re-routing upon network failures.
    - Online (On): TE-Trunks are modified (TE-Trunks resizing, LSPs re-routing, LSPs creation/deletion) according to traffic matrix evolution, or network failure. In such mode, path computation time should be minimized so a to ensure good reactivity.
2) Path Computation Method:
    - Coordinated (Coo): TE-Trunk paths are computed taking into account all TE-Trunks demands in the network.

  - Uncoordinated (Unc): The path(s) of TE-Trunks starting on a given Edge Router are computed without taking into account TE-Trunks originated by other Edge Routers.

3) Function Distribution:
  - Centralized (Cen): The function is located on a single computing element. Figure 2 illustrates an MPLS-TE system based on our architecture where TM, TAG, TC and TAD blocks are centralized and other blocks: TU, IGP-TE and the RSVP-TE are localized on Edge Routers (actually the IGP-TE and the RSVP-TE are localized on all routers).

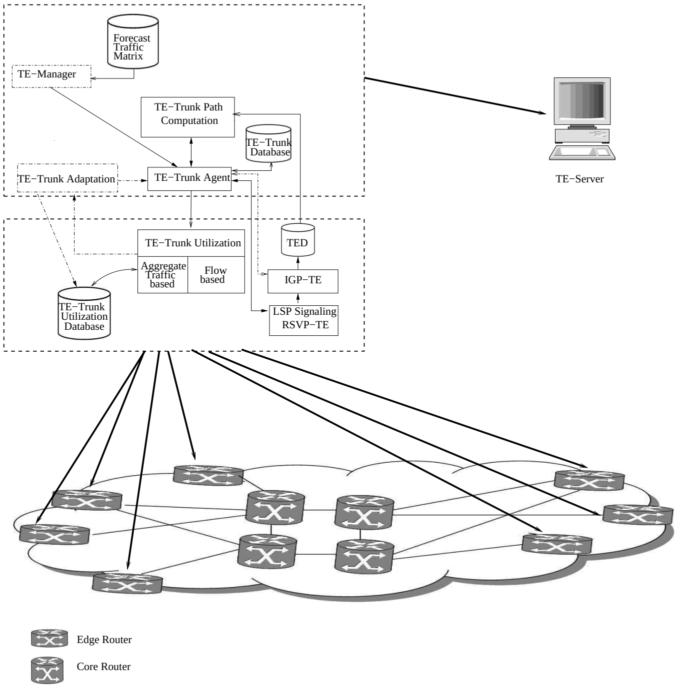

Fig. 2. An Example of a Centralized MPLS-TE System based on the architecture

  - Distributed (Dis): The function is Distributed on multiple computing elements. Figure 3 illustrates an MPLS-TE system based on our architecture where TM is centralized and others blocks: TAG, TC, TAD, TU, IGP-TE and RSVP-TE are distributed.

B. TE evaluation metrics

In order to perform a qualitative evaluation of the efficiency and the applicability of MPLS-TE Routing Systems, a set of metrics are specified :

- Optimality (Opt): The ability to maximize the amount of traffic that can transit in a network with guaranteed QoS. Different performance objectives can be considered such as the residual bandwidth on the most loaded link, the cumulative bandwidth consumption, or, under congestion, the number of rejected requests or the amount of rejected bandwidth.
- Scalability (Sca): The ability to scale well with an increase of any of the following parameters: Number of links/nodes, number of TE-Trunks, number of TE-LSPs, and number of external elements (e.g. PCEs), etc.
- Stability (Sta): The ability to avoid route oscillations and to minimize any perturbation on the network resulting from the establishment of new LSPs (number of signalling messages, message rate, etc.).

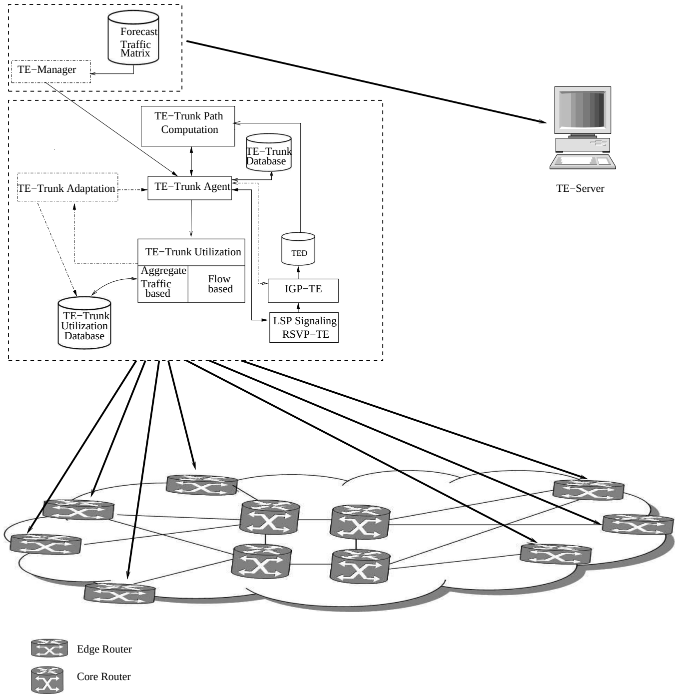

Fig. 3. An Example of a Distributed MPLS-TE System based on the architecture

- Reactivity (Rea): The ability to rapidly react and adapt to a traffic matrix change or/and a topology change. Traffic matrix change implies Trunk suppression/creation/resizing and topology change implies re-routing of traffic on backup paths.

C. Function Distribution

We discuss in this section the distribution of each architecture's function and its impact on the MPLS-TE Systems performances. There are some functions of the architecture which should be only Distributed, others only Centralized on a network server and others which can be either Distributed or Centralized. When two functional blocks that need to communicate, are not located in the same element (e.g. One is located in an Edge Router and the other is located in the TE server) a standard communication protocol is required to manage the communication and the cooperation between the two blocks. In contrast, when these two blocks are located in the same element (e.g. an Edge Router) a communication protocol is not required, such communication may rely on a software interface (e.g. Inter-Process Communication API).

- The MPLS-TE protocols (RSVP-TE and IGP-TE) are Distributed on the routers (note that the IGP-TE may passively run on the Path Computation Block when it is Centralized, so as to feed the TED).
- The TE-Trunk Utilization block should be Distributed as it is in charge of the routing of incoming flows within the TE-Trunks and of TE-Trunk load measurement on Edge Routers. The centralization of this block may affect the reactivity of the MPLS-TE System due to the amount of information to be communicated between Edge Routers and the TE server.
- By definition, the TE-manager is always Centralized.
- The TE-Trunk Agent can either be Centralized on a network

server or Distributed on TE-Trunk Edge Routers. In a Distributed mode, it maintains only TE-Trunks for which it is the head-end. In a Centralized scenario, the TE-Trunk Agent has a global knowledge of all the TE-Trunks. In this case, a communication protocol is required to communicate with Edge Routers (LSP configuration). This may rely for instance on a standard configuration protocol (e.g. XML conf or SNMP). The notification of network failures should be event-driven (e.g. SNMP traps) so as to minimize the amount of information between the Edges Routers and the TE-server.
- The TE-Trunk Adaptation function should always be linked to the TE-Trunk Agent, that is if the TE-Trunk Agent is Centralized (respectively Distributed), the TE-Trunk adaptation is also Centralized (respectively Distributed). When the TE-Trunk Adaptation is Centralized, a communication protocol is required between the TE-Trunk Utilization block located in Edge Routers and the TE-Trunk Adaptation, so as to inform about the LSP load. Such notification should be event-driven so as to minimize the communication between Edge Routers and the TE-server (it is not necessary for the Adaptation block to consult periodically the TE-Trunks Utilization database. The TE-Trunk Utilization block sends a message to the Adaptation block only when a threshold is reached). The separation of these two functions (the TE-Trunk Adaptation and the TE-Trunk Agent) would not bring any value and would require the communication of a lot of information.
- The TE-Trunk Path Computation block may be Distributed or Centralized. (1) If the TE-Trunk Agent is Centralized, the TE-Trunk Path Computation block should also be Centralized (the Coordinated mode) because the separation of these two functions would not bring any value and would require the communication of a lot of information. (2) But, if the TE-Trunk Agent is Distributed, the TE-Trunk Path Computation block may either be Distributed or Centralized. When the TE-Trunk Agent is distributed and the TE-Trunk path Computation is Centralized, the TE-Trunk Path Computation remains Uncoordinated because the TE-Trunk Agents send requests independently. This requires a path computation communication protocol between the TAG and the TC. Such a protocol is under definition within the Path Computation Element (PCE) working group in the IETF [1]. A PCE is defined as an entity that is capable of computing a network path based on a network graph, and applying computational constraints [11]. A PCE serves path computation requests sent by Path Computation Clients (PCCs). The PCE communication Protocol (PCEP) has been defined to support communication between PCCs and PCEs (see [12]). Here the TE-Trunk Agent acts as a PCC and the TE-Trunk Computation block acts as a PCE. Note that the PCE based architecture may also apply when the TE-Trunk Agent and the TE-Trunk Path Computation functions are centralized but not located in the same TE server.

D. Evaluation

By combining the various criteria defined previously and by considering also the Architecture's function distribution discussed in the previous section, a set of MPLS-TE Systems families or approaches are identified and evaluated (the table I summarizes the evaluation results):

1) The On/Dis/Unc MPLS-TE approach: This is an Online Distributed based approach where the requests are handled in a Uncoordinated manner. This approach achieves ”bad” performances in terms of optimality because of its Uncoordinated and Online schemes. In return, as each Edge Router handles only its own requests, it offers ”good” scalability for the MPLS-TE System. However, as the online mode implies LSP creation/deletion/resizing, and the Uncoordinated mode may imply some TE mechanisms such as preemption and crankback, the system is ”poor” in terms of stability. However, according to the reactivity, this approach offers, due to its On/Dis scheme, ”good” performances. In fact, all functions of the MPLS-TE System are located in the same element (the Edge Router) and hence this does not require heavy communication between Edge Router and TE-server.

2) The On/Dis/Coo MPLS-TE approach: This approach operates in a Coordinated manner with other network Edge Routers. Thus, network resources usage is optimized because each Edge Router computes paths by taking into account all TE-Trunks demands in the network but not as well as if the Offline mode was used where there is no time constraint. Hence, it can ensure ”good” performances in terms of optimality. This option offers ”bad” performances in terms of scalability, Edge Routers are likely to be saturated because they have to exchange all the information about their own TE-Trunks/LSPs. This cannot scale because of the number of TE-Trunks in the network and their activity rates (resizing,...). So, it seems not relevant to let all the Edge Routers handle all requests. However, as there is no pre-emption or crankback mechanisms in the Coordinated mode, this approach may result in ”good” performances in terms of stability. But, it cannot ensure ”high” stability because of the Online mode. The ”high” reactivity can be also slightly affected by the Coordinated mode. In fact, in a Coordinated mode, each Edge Router takes into account all TE-Trunk requests to compute TE-Trunks paths which may take potentially long time. Thus the reactivity is affected.

3) The On/Cen/Unc MPLS-TE approach: This approach may achieve ”bad” performances in terms of optimality because of its Uncoordinated and Online schemes. Also, it may suffer from ”poor” scalability performances because of the Centralized mode as the TE-server may not scale with a network size increase. As the online mode implies LSP creation/deletion/resizing, and the Uncoordinated mode may imply some TE mechanisms such as preemption and crankback, the system also suffer from ”poor” stability. This approach also achieves ”poor” performances in terms of reactivity because of the Centralized mode because the TE-server and the Edge Routers should always communicate. For instance, in case of a topology change (network failure), the recovery upon network failure would imply the following sequence: (1)- Failure notification on the TE-server (it may rely on an SNMP trap), (2)- paths computation and (3)- communication of the new paths to all Edge Routers, which may take long time.

4) The On/Cen/Coo MPLS-TE approach: Compared to the previous approach (On/Cen/Unc), this approach improves the optimality and so ensure ”good” performances as we move to a Coordinated mode but it is not highly improved as we are still in an Online mode where the path computation is time constrained. Like the previous approach (On/Cen/Unc), this approach may suffer from ”poor” scalability performances because of the Centralized mode and like the (On/Dis/Coo) approach, it may result in ”good” performances in terms of stability because there is no preemption or crankback mechanisms in the Coordinated mode. But, it cannot ensure ”high” stability because of the Online mode. In terms of reactivity, this approach offers ”bad” performances. For instance, in a network failure

---

[1] Internet Engineering Task Force

case, the recovery would imply the following sequence: (1)- Failure discovery on the TE-server, (2)- Coordinated path computation and (3)- communication of the new paths to all Edge Routers, which may take potentially long time.

5) The Off/Dis/Unc MPLS-TE approach: This approach adopts the Offline mode where TE-Trunks paths, potentially including Backup paths, are pre-computed periodically without real time computation constraints, but it remains ”poor” in terms of optimality due to its Dis/Unc scheme. In return, this scheme allows to avoid the message exchange between the TE-server and the Edge Routers and to accomplish ”good” scalability performances as each Edge Router handle only its own TE-Trunk requests. According to the stability, this approach can achieve ”high” performances as there is no TE-Trunk Adaptation (Trunk suppression/creation/resizing) and LSP re-routing. All Offline approaches are by definition not reactive (”bad” reactivity) and stable (”high” stability).

6) The Off/Dis/Coo MPLS-TE approach: This approach can achieve ”high” performances in terms of optimality with its Off/Coo because TE-Trunk paths are computed in Offline mode taking into account all TE-Trunks demands in the network which offers a global network optimization with no time constraint. In contrast, this approach offers ”bad” performances in terms of scalability because of its Dis/Coo scheme as each Edge Router has to maintain all the TE-Trunk requests.

7) The Off/Cen/Unc MPLS-TE approach: As the Offline mode is adopted, the performances of the MPLS-TE system in terms of Optimality may be improved. But The Uncoordinated scheme of this approach affects these performances. So, like the Off/Dis/Unc approach, this approach suffers from ”poor” optimality performances. This approach may be ”poor” in terms of scalability because of the Centralized mode as the TE-server may not scale with network size increase (node number, etc.).

8) The Off/Cen/Coo MPLS-TE approach: This approach can ensure ”high” performances in terms of optimality. In fact, TE-Trunks placement can be drastically optimized because the TE-Trunk Path Computation function knows all the requests and can perform a Coordinated path computation, with no time limitation. Like the previous approach (Off/Cen/Unc), this approach may be ”poor” in terms of scalability due to its Centralized scheme.

As shown in the previous sections, TE criteria either alone or combined can influence the performances of a MPLS-TE System:

- An approach which adopts **Centralized** scenario, may suffer from **scalability** and **reactivity** issues.
- An approach which adopts **Online** mode may suffer from network **stability** problems.
- An approach which adopts **Offline** mode lacks in **reactivity**.
- An approach which adopts a **Dis/Coo** scenario faces **scalability** problems.
- An approach which adopts a **Dis/Unc** scenario affects the performances of an MPLS-TE System in terms of **optimality**.

The result of this qualitative evaluation is that special care must be taken when combining different TE approaches to build an MPLS-TE Routing System.

IV. CONCLUSION

MPLS-TE is being deployed by network operators to better optimize their network resources. The routing in MPLS-TE networks is a large and open issue. Studies aimed to improve MPLS-TE routing in terms of scalability, stability, robustness, optimality and survivability. In this paper, we have proposed a generic architecture for MPLS-TE Routing Systems, that combines MPLS-TE functional blocks such as TE-Trunk Computation, TE-Trunk utilization and TE-Trunk Adaptation. This generic architecture is proposed to facilitate the classification of MPLS-TE Routing solutions, to improve existing mechanisms and to propose new solutions. By relying on this architecture, we can identify and evaluate a set of MPLS-TE Routing approaches using several evaluation metrics. We have showed that some TE criteria, either alone or combined, can influence the performances of a MPLS-TE Routing System.

TABLE I
EVALUATION OF DIFFERENT MPLS-TE APPROACH COMBINATIONS

	Optimality	Scalability	Stability	Reactivity
On/Dis/Unc	–	**	*	* * *
On/Dis/Coo	**	–	**	**
On/Cen/Unc	–	*	*	*
On/Cen/Coo	**	*	**	*
Off/Dis/Unc	*	**	* * *	–
Off/Dis/Coo	* * *	–	* * *	–
Off/Cen/Unc	*	*	* * *	–
Off/Cen/Coo	* * *	*	* * *	–

(”* * *” high, ”**” good, ”*” poor, ”–” bad)

REFERENCES

[1] D. Awduche, J. Malcolm, J.Agogbua, M. O’DEll, and J. McManus, “Requirements for traffic engineering over MPLS, rfc 2702,” September 1999.

[2] H. Smit and T. Li, “IS-IS extensions for traffic engineering, rfc 3784,” June 2004.

[3] D. Katz, K. Kompella, and D. Yeung, “Traffic engineering (TE) extensions to ospf version 2.”

[4] D. Awduche, L. Berger, D. Gan, T. Li, V. Srinivasan, and G. Swallow, “RSVP-TE: Extensions to rsvp for LSP tunnels, rfc 3209,” December 2001.

[5] T. L. M. Kodialam, “Minimum interference routing with applications to MPLS traffic engineering,” *IEEE INFOCOM*, 2000.

[6] W. Szeto, R. Boutaba, and Y. Iraqi, “Dynamic online routing algorithm for MPLS traffic engineering,” *NETWORKING 2002*, pp. 936–946.

[7] P. W. S. Suri, M. Waldvogel, “Profile-based routing: A new framework for MPLS traffic engineering.”

[8] K.-T. K. S. Phuvoravan, “Fast time scale control for MPLS traffic engineering,” April 2002.

[9] R. Battiti and E. Salvadori, “A load balancing scheme for congestion control in MPLS networks,” November 2002.

[10] A. Bosco, “Edge distributed admission control in MPLS networks,” *Communications Letters, IEEE Volume 7, Issue 2*, pp. 88–90, Febrary 2003.

[11] A. Farrel, J. Vasseur, and J. Ash, “Path computation element (PCE) architecture, IETF draft,” July, 2005.

[12] J. Vasseur, J. L. Roux, A. Ayyangar, E. Oki, Y. Ikejiri, A. Atlas, and A. Dolganow, “Path computation element (PCE) communication protocol (PCEP),ietf draft,” November 2005.