

Towards an Adaptive and Intelligent MPLS Network

Rana Rahim-Amoud, Leila Merghem-Boulahia, and Dominique Gaiti
ISTIT, University of Technology of Troyes
12, rue Marie Curie, BP 2060, 10 010 TROYES CEDEX, FRANCE
{rana.amoud, leila.boulahia, dominique.gaiti}@utt.fr,
WWW home page: <http://www.utt.fr/labos/LM2S/>

Abstract. The Multi-Protocol Label Switching (MPLS) is an Internet Engineering Task Force (IETF) framework. It is a versatile solution to address the problems faced by present-day networks like speed, scalability and traffic engineering. However, the Quality of Service (QoS) management of MPLS is made by static methods. In this paper, we propose a solution based on Multi-Agent Systems (MAS) to manage the QoS into MPLS by adequate adaptive methods.

1 Introduction

The Multi-Protocol Label Switching (MPLS) [10] is a new technology that integrates the label-swapping paradigm with network-layer routing. This means that it uses labels, instead of IP addresses, to forward packets.

MPLS network provides more advantages than the traditional IP networks like the Traffic Engineering (TE) [3]. With the use of TE, MPLS provides a considerable efficient traffic control [11] more than the traditional OSPF or IS-IS. In addition, MPLS provides the Virtual Private Networks [5] which are currently the most selected by a wide range of costumers to provide private services, etc. Many protocols can be used to distribute labels, LDP (Label Distribution Protocol) [1], CR-LDP (Constraint based-LDP) [2] and RSVP-TE [4]. With these previous features, the MPLS management is not as simple as a traditional management function.

As network conditions frequently change, the network must be able to recognize and adapt itself to the running traffic conditions in order to choose the most appropriate algorithm to be used.

In line with this, recent researches showed the effectiveness of Multi-Agent System (MAS) for the dynamic management of distributed systems. This explains our tendency to use a Multi-Agent System to introduce some decision-making abilities in the complex network management tasks.

This paper is organized as follows. We first find the decision points into MPLS, and then we present our proposed architecture. Finally, we make a brief conclusion.

2 The actual weakness of MPLS

Currently, the actual weakness of MPLS resides in its inability to provide application-level routing intelligence, which is a fundamental component especially for voice delivery [9].

Voice over IP (VoIP) is a critical application that requires intelligent routing alternation on the call level to prevent latency, delay, packet loss and jitter and this cannot be provided by MPLS.

Adapting MPLS to VoIP traffic necessitates the distinction of different traffic types. Some solutions, which couple MPLS with DiffServ or RSVP, were proposed to solve MPLS limitations.

DiffServ over MPLS, according to [9], is able to provide end-to-end solution for MPLS only if the entire traversed path is controlled by a single entity. Most importantly, "DiffServ does nothing to solve the static route problem of MPLS" [9] while it introduces complexity into the architecture.

The implementation of RSVP with MPLS enables the reservation of bandwidth on a router for each LSP. In this way, it is possible to prevent an overbooking in the router from the start. However it does nothing to solve the fluctuating demands of VoIP.

Even with these proposed solutions, MPLS remains unable to guarantee the QoS of incoming traffic, so it is very essential to find another solution. Our proposition is based on the introduction of an MAS within the MPLS domain, to do that, we must first find the positions in which the intelligent agents will be introduced. These positions are the MPLS decision points.

3 MPLS Decision points

One of the most important steps of our research is to find the MPLS decision points. Once found, we can add our agents within these points.

3.1 The first MPLS decision point

Different types of traffic flows arrive to the entry of the MPLS domain. The classification of the packets is done just at the entry of the domain by the I-LSR (Ingress-Label Switch Router), by assigning a particular packet to a particular FEC (Forwarding Equivalence Class). Within the domain, there is no reclassification, packets are just switched.

Currently the most used criteria to build FEC is based on the prefix or the destination address, by taking advantage from the aggregation of flows that have the same destination. Aggregation may reduce the number of labels which are needed to

handle a particular set of packets. However, it does not take into account the type of traffic, and degrades as a result the network performance.

We propose to introduce an intelligent agent on the level of each I-LSR router which is an efficient and a pertinent decision point. This agent will have as a role to examine the incoming flows and to create for each type of traffic a different FEC and consequently a different LSP, even for traffics which have the same destination.

As each E-LSR (Egress-LSR) is at the same time an I-LSR for the packets forwarded in the opposite direction, then our agent will be introduced on the level of each LER (Label Edge Router) within the MPLS domain, and consequently we obtain a Multi-Agent System (Fig .1).

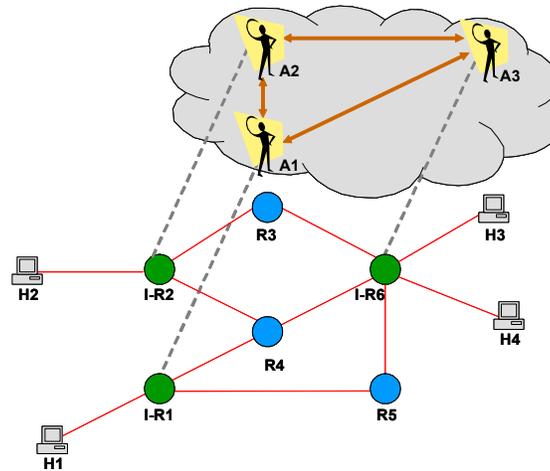


Fig. 1. Introducing an MAS in an MPLS domain

3.2 The second MPLS decision point

According to [6], an LSR is capable of label merging if it can receive two packets from different incoming interfaces, and/or with different labels, and send both packets on the same outgoing interface with the same label.

In the following example (Fig. 2), if LSR R6 was merge-capable, it simply would act as an E-LSR to upstream neighbors and as Merge LSR to downstream neighbors [6]. Then, it would perform aggregation of requests from upstream neighbors R1 and R2, reducing as a result the label consumption within the MPLS Network. To do the aggregation, flows must be in the same FEC.

Furthermore, label merging causes a small delay because it is designed for IP traffic and need not to be used for delay-sensitive traffic.

Our proposal is to introduce an agent on the level of the LSR R6 having for role to activate the merge-capable of R6 when traffic is not delay-sensitive and to deactivate it in other cases. As a consequence, the two LSP will be aggregated into only one LSP from their point of intersection.

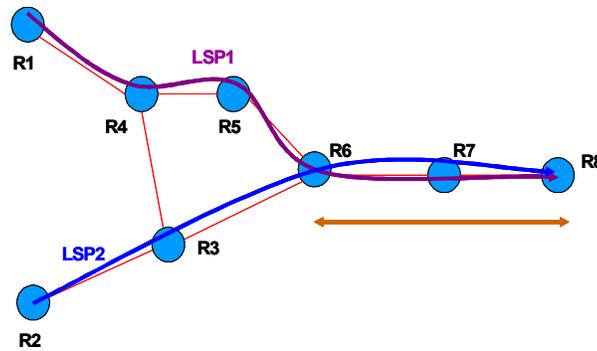


Fig. 2. LSP attribution before the introduction of the intelligent agent

4 MPLS Multi-Agent System Model

Our approach is based on the architecture developed in [7]. It is two-level architecture: The level 1 (Adaptation Level) contains the Master agent which monitors the agents of level 0 (Activated Protocols Level) (Fig. 3).

The most critical issues in MPLS are the creation of LSP, the aggregation of flows, the distribution of these flows to LSP taking into account their natures, etc. According to this, the level 0 will be composed of several agents: LSP Creator Agent, Aggregator Agent, etc (Fig. 3).

The Master Agent observes the current router conditions and chooses the most appropriate protocols to the other agents under its responsibility. Each agent of level 0 has a set of protocols dedicated to the task it is in charge of. For example, LSP Creator agent establishes some rules for selecting which label distribution protocol to use under which circumstances.

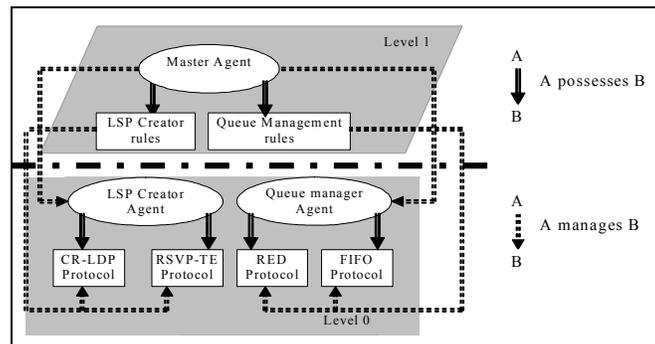


Fig. 3. Two-level MAS architecture

Each Master Agent possesses a set of rules allowing to select the appropriate protocols to activate, and therefore to select the best actions to execute. These rules

respond to a set of events and trigger the actions which affect the protocols supervised by that Master Agent.

Their role is to manage a set of protocols in order to provide the best functioning of the node and to avoid incoherent decisions within the same node. These rules give the node the means to guarantee that the set of actions executed, at every moment, by its agents are coherent, in addition to be the most relevant to the current situation [8]. In order to minimize conflict situations, rules are organized in separate modules following the task they are interested in (Fig. 3 shows an example).

The Master Agent actions adapt the node's protocols and may consist in: letting the protocol running, modifying the internal functioning of the protocol, inhibiting the protocol, activating the protocol.

The actions undertaken by the node have local consequences but may influence the decisions of the other nodes. In fact, by sending messages bringing new information on the sender node's state, a receiver's Master Agent rule may be triggered. This can involve a change within the receiver node (the inhibition of an activated protocol, or the activation of another one, etc.). This change may have repercussions on other nodes, and so forth until the entire network becomes affected.

This dynamic process aims to adapt the network to new conditions and takes advantage of the agents' abilities to alleviate the global system. We argue that these agents will achieve an optimal adaptive management process because of the following two points: (1) each agent holds different processes (protocols and adaptive selection of these protocols) allowing to take the most relevant decision at every moment; (2) the agents are implicitly cooperative in the sense that they possess rules that take account of the neighbors' state in the process of protocols' selection.

5 Conclusions and future work

This paper presents the first step towards an adaptive and intelligent MPLS network. As QoS management configuration may not be suitable for all traffic scenarios, a solution is required in order to respond to this challenge, improve the network performance and satisfy users' requests. Our proposed solution is based on Multi-Agent Systems. Future work in this area should explicitly address the agents' rules in our two-level model improving as a result our architecture. Once defined we will move to make a testbed.

Finance: PhD research supported in part by "Conseil Régional Champagne-Ardenne" (district grant) and the European Social Fund.

References

1. Anderson L, Doolan P, Feldman N, Fredette A, Thomas B (2001) LDP Specification. Network Working Group, RFC3036
2. Ash J, Girish M, Gray E, Jamoussi B, Wright G (2002) Applicability Statement for CR-LDP. Network Working Group, RFC3036

3. Awduche D O (1999) MPLS and traffic engineering in IP networks, IEEE Communications Magazine, vol. 37, no.12, pp. 42-47
4. Awduche D, Berger L, Gan D, Li T, Srinivaan V, Swallow G (2001) RSVP-TE Extensions to RSVP for LSP Tunnels. Network Working Group, RFC3209
5. Carter SF (2005) Quality of service in BT's MPLS-VPN platform. BT TECHNOLOGY JOURNAL 23 (2): pp. 61-72
6. Fredette A, White C, Andersson L, Doolan P (1997) Stream Aggregation. Work in progress, draft-fredette-mpls-aggregation-00.txt
7. Merghem L, Lecarpentier H (2002) Agents: A Solution for Telecommunication Network Simulation. NetCon'2002. Kluwer Academic publishers, pp.165-176. Paris, France.
8. Merghem, L., Gaiti, D., Pujolle, G.: On Using Agents in End to End Adaptive Monitoring. E2EMon Workshop, in conjunction with MMNS'2003. Lecture Notes in Computer Science, Vol. 2839. Belfast, Northern Ireland (2003) 422-435
9. PRIMEDIA Business Magazines & Media Inc., available at: http://infocus.telephonyonline.com/ar/telecom_questioning_mpls/ (2003)
10. Rosen E, Viswanathan A, Callon R (2001) Multiprotocol Label Switching Architecture. RFC3031, IETF
11. Xiao X., Hannan A., Bailey B. et Ni L., « Traffic engineering with MPLS in the Internet », IEEE Network Magazine, pp. 28-33, Mar. 2000.