

# Intelligent Routers and Smart Protocols

Guy Pujolle<sup>1</sup> and Dominique Gai ti<sup>2</sup>

<sup>1</sup> LIP6, University of Paris 6, 75015 Paris, France  
Guy.Pujolle@lip6.fr

<sup>2</sup> UTT, 12 rue Marie Curie, 10000 Troyes, France  
Dominique.Gaiti@utt.fr

**Abstract.** IP networks are now well established. However, control, management and optimization schemes are provided in a static and basic way. Network control and management with intelligent software agents offers a new way to master quality of service, security and mobility management. This new paradigm allows a dynamic and intelligent control of the equipment in a local manner, a global network control in a cooperative manner, a more autonomous network management, and a better guaranty of all important functionalities like end to end quality of service and security. In this paper we provide an illustration of such a paradigm through a testbed of an architecture based on intelligent routers and smart protocols. This Goal-Based Networking (GBN) architecture, using adaptable protocols named STP/SP (Smart Transport Protocol/Smart Protocol), is able to optimize the communications through the networks. Finally, we discuss the pros and cons of this new architecture.

## 1 Introduction

The popularity of the Internet has caused the traffic on the Internet to grow drastically every year for the last several years. It has also spurred the emergence of the quality of service (QoS) for Internet Protocol (IP) to support multimedia application like ToIP. To sustain growth, the IP world needs to provide new technologies for guarantying quality of service. Integrated services and differentiated services have been normalized to support multimedia applications. The routers in the IP networks play a critical role in providing these services. The demand of QOS on private enterprise networks has also been growing rapidly. These networks face significant bandwidth challenges as new application types, especially desktop applications uniting voice, video, and data traffic need to be delivered on the network infrastructure. This growth in IP traffic is beginning to stress the traditional software and hardware-based design of current-day routers and as a result has created new challenges for router design.

To achieve high-throughput and quality of service, high-performance software and hardware together with large memories were required. Fortunately, many changes in technology (both networking and silicon) have changed the landscape for implementing high-speed routers. However, scalability problems were discovered with InterServ technologies and statistical problems with DiffServ. Moreover, these technologies are rather complicated to size and we assist to important configuration problems that need specialized engineers.

This paper proposes a new paradigm for providing a smart networking technique allowing a real time network configuration. Indeed, we propose to introduce intelligent routers able to configure themselves depending on the state of the network and to define a new generation of smart protocols.

The rest of the paper is organized as follows. First we introduce the smart networking paradigm and the implication on the routers. Then, we introduce a new protocol stack, the STP/SP model, followed by the description of the smart architecture (Goal Based Networking architecture) to support the deployment of the intelligent routers and the STP/SP model. Finally, we present an analysis of this architecture and we conclude this work.

## 2 Smart Networking and Intelligent Routers

As user needs are becoming increasingly various, demanding and customized, IP networks and more generally telecommunication networks have to evolve in order to satisfy these requirements. That is, a network has to integrate more quality of service, mobility, dynamicity, service adaptation, etc. This evolution will make users satisfied, but it will surely create more complexity in the network generating difficulties in the control process.

Since there is no control mechanism which gives optimal performance whatever the network conditions are, we argue that an adaptive and dynamic selection of control mechanisms, taking into account the current traffic situation, is able to optimize the network resources uses and to come up to a more important number of user expectations associated with QoS. To realize such functionalities, it is necessary to be able to configure automatically the network in real time. Therefore, all the routers must be able to react to any kind of change in the network. Different techniques could be applied but as the most difficult moment is congestion, the technique has to be autonomic and routers have to turn into intelligent routers.

Due to these different issues, a multi-agent approach is the solution. In fact, agents own some features like autonomy, proactivity, cooperation, etc. predisposing them to operate actively in a dynamic environment like IP networks. Agents, by consulting their local knowledge and by taking into consideration the limited available information they possess about their neighbors, select the most relevant management mechanisms to the current situation.

A multi-agent system is composed of a set of agents which solve problems that are beyond their individual capabilities [1]. Multi-agent systems have proven their reliability when being used in numerous areas like: (1) the road traffic control ([2], [3]); (2) biologic phenomena simulation like the study of eco-systems [4] or the study of ant-colonies [5], for example; (3) social phenomena simulation like the study of consumer behaviors in a competitive market [6]; (4) industrial applications like the control of electrical power distribution systems, the negotiation of brands, etc. By its nature, multi-agent approach is well suited to control distributed systems. IP networks are good examples of such distributed systems. This explains partly the considerable contribution of agent technology when introduced in this area. The

aim was mainly to solve a particular problem or a set of problems in networks like: the discovery of topology in a dynamic network by mobile agents ([7], [8]), the optimization of routing process in a constellation of satellites [9], the fault location by ant agents [10], and even the maximization of channel assignment in a cellular network [11].

Our approach consists in integrating agents in the different routers. These agents optimize the network QoS parameters (delay, jitter, loss percentage of a class of traffic, etc.), by adapting the activated control mechanisms in order to better fit the traffic nature and volume, and the user profiles. Agents may be reactive, cognitive or hybrid [1], [4], [12]. Reactive agents are suitable for situations where we need less treatment and faster actions. Cognitive agents, on the other side, allow making decisions and planning based on deliberations taking into account the knowledge of the agent about itself and the others. A hybrid agent is composed of several concurrent layers. In INTERRAP [13], for example, three layers are present: a reactive layer, a local planning layer, and a cooperative layer. The approach we propose is different. In fact, every node has one cognitive agent that supervises, monitors, and manages a set of reactive agents. Each reactive agent has a specific functioning realizing a given task (queue control, scheduling, dropping, metering, etc.) and aiming to optimize some QoS parameters. The cognitive agent (we call it Master Agent) is responsible for the control mechanisms selection of the different reactive agents, regarding the current situation and the occurring events. By using such an architecture, we aim to take advantage of both the reactive and cognitive approaches and avoid shortcomings of the hybrid approach (coordination between the different layers, for instance).

To get the agent-based smart networking approach, we propose to select the appropriate control mechanisms among:

- Adaptive: the agent adapts its actions according to the incoming events and to its vision of the current system state. The approach we propose is adaptive as the agent adapts the current control mechanisms and the actions undertaken when a certain event occurs. The actions the control mechanism executes may become no longer valid and must therefore be replaced by other actions. These new actions are, indeed, more suitable to the current observed state;
- Distributed: each agent is responsible for a local control. There is no centralization of the information collected by the different agents, and the decisions the agent performs are in no way based on global parameters. This feature is very important as it avoids having bottlenecks around a central control entity;
- Local: the agent executes actions on the elements of the node it belongs to. These actions depend on local parameters. However, the agent can use information sent by its neighbors to adapt the activated control mechanisms;
- Scalable: our approach is scalable because it is based on a multi-agent system which scales well with the growing size of the controlled network. In order to adaptively control a new node, one has to integrate an agent (or a group of agents) in this node to perform the control.

Our model relies on two kinds of agents: (1) Master agent: which supervises the other agents in addition to what is happening in the node; (2) the other agents: which are responsible for a specific management task within the node. We can distinguish the two following levels of decision within a node:

At level 0, we find the different control mechanisms of the node, which are currently activated. Each control mechanism is characterized by its own parameters, conditions and actions, which can be monitored and modified by the Master Agent. Some of the proposed management mechanisms are inspired from known algorithms but have been agentified in order to get better performance and better cooperation between agents.

Different agents belong to this level (Scheduler Agent, Queue Control Agent, Admission Controller Agent, Routing Agent, Dropping Agent, Metering Agent, Classifying Agent, etc.). Each of these agents is responsible for a specific task within the node. So each agent responds to a limited set of events and performs actions ignoring the treatments handled by other agents lying on the same node or on the neighborhood. This allows to the agents of this level to remain simple and fast. More complex treatments are indeed left to the Master Agent.

At level 1, is lying a Master Agent responsible for monitoring, managing, and controlling the entities of level 0 in addition to the different interactions with the other nodes like cooperation, negotiation, messages processing, etc. This agent owns a model of its local environment (its neighbors) that helps him to take its own decisions. The Master Agent chooses the actions to undertake by consulting the current state of the system (neighbors nodes state, percentage of local loss, percentage of its queue load, etc.) and the meta-rules at its disposal in order to have only the most relevant control mechanisms activated with the appropriate parameters. The node, thanks to the two decision levels, responds to internal events (loss percentage for a class of traffic, load percentage of a queue, etc.) and to external ones (message sent by a neighbor node, reception of a new packet, etc.).

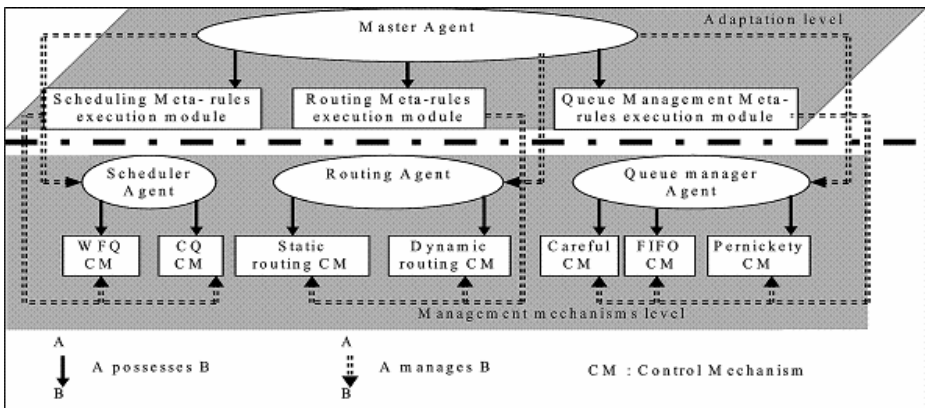


Fig. 1. Two levels of decision within the node

The Master Agent owns a set of meta-rules allowing it to decide on actions to perform relating to the different node tasks like queue management, scheduling, etc. (Figure 1). These meta-rules permit the selection of the appropriate control mechanisms to activate the best actions to execute. They respond to a set of events and trigger actions affecting the control mechanisms supervised by that Master Agent. Their role is to control a set of mechanisms in order to provide the best functioning of the node and to avoid incoherent decisions within the same node. These meta-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.

The actions of the routers have local consequences in that they modify some aspects of the functioning of the router (its control mechanisms) and some parameters of the control mechanisms (queue load, loss percentage, etc.). They may, however, influence the decisions of other nodes. In fact, by sending messages bringing new information on the state of the sender node, a Master Agent meta-rule on the receiver node may fire. This can involve a change within the receiver node (the inhibition of an activated control mechanism, 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 to take advantage of the agent abilities to alleviate the global system. We argue that these agents will achieve an optimal adaptive control process because of the following two points: (1) each agent holds different processes (control mechanisms and adaptive selection of these mechanisms) allowing to take the most relevant decision at every moment; (2) the agents are implicitly cooperative in the sense that they own meta-rules that take into account the state of the neighbors in the process of control mechanisms selection. In fact, when having to decide on control mechanisms to adopt, the node takes into consideration the information received from other nodes.

### 3 A New Smart Architecture STP/SP

In the previous section we introduced intelligent routers. In this section we are interested in discussing the opportunity to link the intelligent routers using smart protocols adapted to the environment and the type of traffic.

TCP/IP architecture was created for the interconnection of networks running with different architectures. Then, the TCP/IP architecture was chosen as the unique architecture for all communications. The advantage is clearly to permit a universal interconnection scheme of any kind of machines. However, TCP/IP is only a trade-off and we wonder if specific architectures IP compatible or not could not be a better solution to optimize the communications. It was shown in paper [14] that TCP/IP is not the optimum protocol as soon as some constraints have to be realized. For example, TCP/IP is a rather bad protocol for energy consumption and not at all adapted to sensor networks.

The idea is to propose a Smart Protocol (SP) that can adapt to the environment, for optimizing battery or optimizing reliability or optimizing QoS or any other interesting functionality. The design of a Smart Protocol at the network layer that is aware of the upper and the lower layers and adapts their communication to a set of

parameters is obviously the ultimate communication architecture that can support current and emerging wireless networks. This new context-aware architecture that we named STP/SP Smart Transport Protocol/Smart Protocol could be compatible with IP.

Indeed, the SP protocol is a set of protocols SP1, SP2, ...SPn that could be either derived from the IP protocol or could be adapted to specific environments. In the same way the STP protocol is a set of protocol that could be derived from the TCP protocol or from independent protocols. In this paper, we are interested in the compatibility of STP/SP with the TCP/IP architecture. Indeed, the TCP/IP functionalities are rich enough to cope with the different situations.

All the different architectures are easily interconnected through a classical TCP/IP protocol. For instance, a sensor network will deploy its STP/SP protocol stack that support the requirements of the application set up over the sensor network. This sensor network will be interconnected through a classical TCP/IP gateway to another network that deploys another STP/SP protocol stack which supports the requirements of this other network. This might sound as going back to the period where the networks deploy their proprietary protocols. Then, IP was designed to interconnect these networks. Next IP was generalized and today reached the point where this protocol cannot cope with all types of environment such as wireless environments. The difference between the STP/SP approach and the former proprietary solutions is that STP/SP will basically use the TCP/IP concepts and functionalities, but in a smart way. In fact, rather than deploying TCP/IP in the same way in any environment without being aware of the requirements of this environment, STP/SP will offer a smart TCP/IP like environment. This will keep the simplicity and efficiency of TCP/IP, but will add a smart process that is totally absent in TCP/IP. This smart process will be deployed using a new architecture in the network guided by a set of objectives named Goals.

We describe this global architecture in Figure 2. The objective of this architecture is to implement the smart process of selecting the sub-protocol of the STP/SP protocol that fulfils the requirements of the concerned network. This is a goal-based networking architecture and the control is a goal-based control.

## 4 A Goal-Based Networking Architecture

The goal-based architecture is composed of mainly two mechanisms: The smart mechanism to select the STP/SP protocol and its parameters, and the enforcement mechanism to enforce the decisions of the smart mechanism. For that we use the agent-based scheme described in the previous section, and we use some concepts of the policy based networking [15] such as the enforcement procedures to implement the mechanism.

An agent-based platform permits a meta-control structure such as the platform described in [16]. Assuming that for each network node we associate one or several agents, the network can be seen as a multi-agent system. The main goal of this system is to decide about the control to use for optimizing a given functionality described in the goal distributed by the Master Agent.

Intelligent agents are able to acquire and to process information about situations that are "not here and not now", i.e., spatially and temporally remote. By doing so, an agent may have a chance to avoid future problems or at least to reduce the effects. These capabilities allow agents to adapt their behavior according to the traffic flows going through the node.

It is important to note that other works has proposed a decision mechanism in the network to enforce decision or policies in the network. This typical architecture named Policy-based Networking (PBN) enforces high level decisions without unfortunately considering the problem optimization of parameters related to lower levels of the network. It's only a top down approach. In our proposed architecture, we intend to use the enforcement procedure of policy-based networking architecture that is an interesting concept for automating the enforcement of the smart mechanism decisions. The Goal-based architecture considers the optimizing problem related to the higher but also the lower layers of the network, and enforces the most suitable STP/SP protocols and parameters for the given network and application.

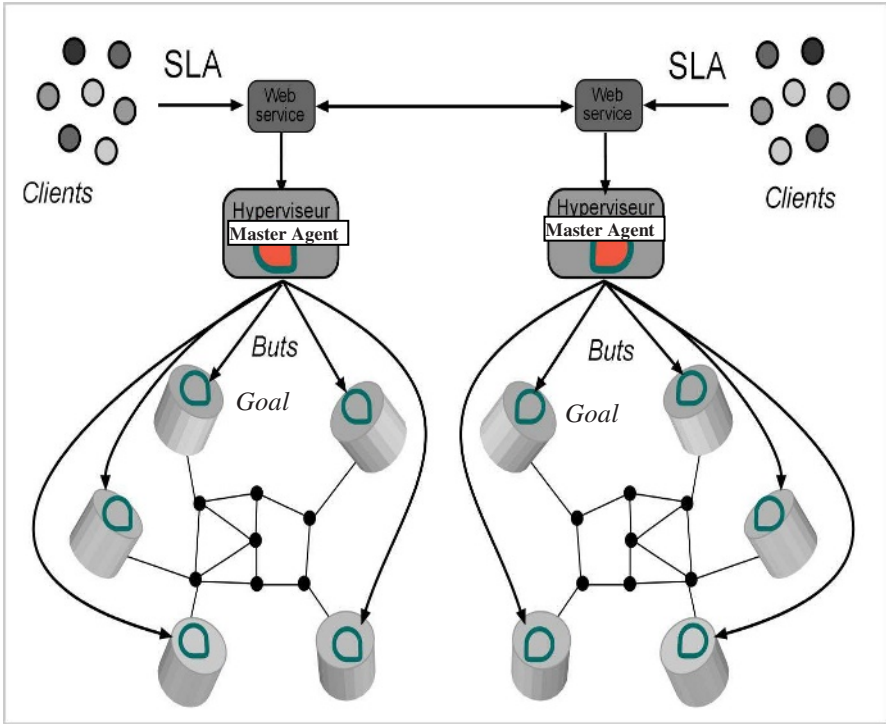


Fig. 2. The global Goal-based Networking architecture

Figure 2 depicts the global Goal-based Networking architecture (GBN) and Figure 3 depicts the GBN and STP/SP reference model.



First, users can enter their SLA through a Web service scheme for example. The manager of the network can also enter the network configurations corresponding to the goals of the network. A Master Agent in layer 1 is able to decide about the global goal of the network. This Master Agent is supported by any kind of centralized servers if any. As soon as defined, the goal is distributed to the different routers that could be named Goal Enforcement Point (GEP). Knowing the goal, the different nodes have to apply policies and define the control mechanisms. A configuration of the routers is provided to reach the goal. The configuration affects the software, the hardware and the protocol stack.

The agents in the GEP are forming the multi-agent system of level 0 described in the previous section.

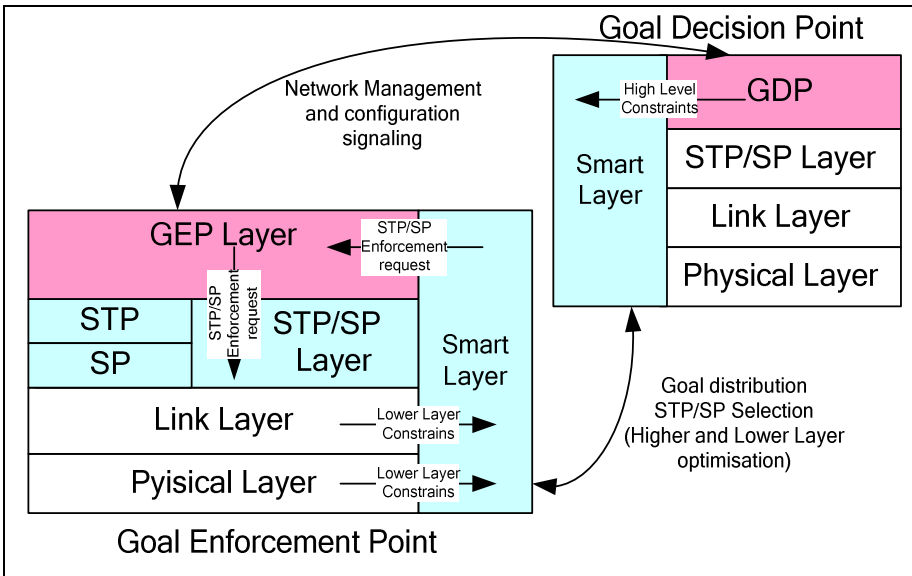


Fig. 3. GBN and STP/SP reference Model

The Smart Layer is in charge of collecting the different constraints from the layers 0 but also from the higher layer (layer 1), then specify and update the goal of the network which is about what to optimise in the network and what to be offered by the network. Note that the classical approaches consider only, what to be offered by the network. After specifying the network goal, the smart layer selects the STP/SP protocols and parameters that will optimize the specified goal. The smart layer will keep updating the goal of the network based on the current state of the network or on a new policies introduced by the Goal Decision Point.

The choice of the protocol can be seen at two levels: the local (level 0) and the global level (level 1). One specific agent in each node (Smart Layer) may be defined for deciding the local protocol in cooperation with the other similar agent of the multi-agent system. Each agent has to perform a specific procedure, which is



triggered according to the state of the node, to the QoS required, and to any other reason. This constitutes a local level for the decision. Moreover, agents can periodically interact to exchange their knowledge and ask to other agents if they need information they do not have. This constitutes the global level.

The smart layer interacts with the Goal Enforcement point (GEP) in order to enforce the STP/SP selected protocol that realizes the global goal. This implies also the definition of the algorithms to manage the CPU, the sensor, the radio or any parameter of the traffic conditioner as shown in section 3.

Indeed, the traffic conditioner is replaced by an extended traffic conditioner (XTC) where different algorithms can be supported. The GEP is in charge to decide the value of the parameters and to decide about the protocol to be used. Within the entities that can be configured, classical control mechanisms as droppers, meters, schedulers, markers, etc. may be found but also resource of the battery, availability, security parameters, radio parameters, etc. This XTC is shown in Figure 4.

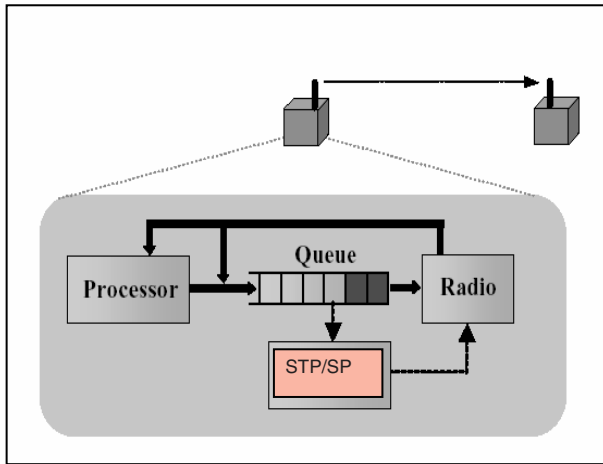


Fig. 4. The Extended Traffic Conditioner XTC

## 5 Simulation and Testbed Results

In this section, we are interested in a performance evaluation of a simple testbed to understand the pros and the cons of the new architecture with intelligent routers and a smart protocol stack.

For the STP/SP architecture we chose only two states for the SP protocol: a protocol using packets as long as possible and a protocol with only short packets (100 bytes). Two kinds of clients were defined:

- Telephony which induces an IP packet payload of 16 bytes and a throughput of 8 Kbps per call. The IP packet may be either padded to reach 100 bytes or can group several available payloads. In this case the waiting time cannot exceed 48 ms (namely three payloads can be encapsulated in the same IP packet). The

response time of the end to end delay cannot be greater than 200 ms and only 1 percent of packets may arrive in late (they are dropped at the arrival but the quality of the voice is maintained).

- File transfer with 1 million bytes per file. When available packet get a 10 000 bytes length and in the other case the file is segmented to produce 100 bytes packet length.

The arrival process of telephone calls is exponentially distributed. The length of telephone calls is 3 minutes on the average and exponentially distributed. The arrival process of the files transfers is exponentially distributed and the average length is 1 million bytes at a constant rate of 2 Mbps. Traffics introduced by these two applications are identical and equal to 1 Mbps. Namely, idle period and busy period for the file transfer are 0.5. On the average 125 telephone calls are running.

Two goals were defined: minimizing the energy consumption in the global network and optimizing the number of successful telephone calls.

The model is a tandem queuing system composed of five nodes in series. The first queue receives the arriving packets and the queues are FIFO. The service process is dependent on the length of the packets with a rate of 2.5 Mbps.

Results of our simulation show that the lifetime of the networks is more than twice when the length of the packets is as long as possible but 20% of the telephone calls are losing more than 1 percent of packets so are dropped. The energy consumption is divided by more than two. On the contrary, when using 100 bytes length packets, all the telephone calls are running correctly but the lifetime is divided by 2.

The previous example does not take into account the possibility to add intelligent routers in the network. To analyze the performance of intelligent routers we developed a simulation package that includes a simulation of networks elements as routers, switches, terminal equipment and so on and the real agent-based software with the Master Agent and the real time agents at the layer 0. Today a large number of results are available in different papers showing the efficiency of the method.

The main drawback of this solution is the fact to add a large number of software agents in the network and increase the complexity of the routers. Indeed, this complexity is quite easy to handle with the new generation of routers offering a JVM through a standard interface. All the mechanisms described are now under industrial development via a start-up depending on Paris 6 University and the University of Troyes.

## 6 Conclusion

This paper introduced a new communication architecture to better support QoS and new functionalities using intelligent routers and smart protocols. Intelligent routers are self configurable using an agent-based control scheme. STP/SP (Smart Transport Protocol/Smart Protocol) is a smart communication model that will use different transport and network protocols adapted to the current environment. This architecture and these protocols consider not only the policies provided by

the business plan but also the constraints of the lower layers of the network. A Goal-based architecture is proposed to provide the selection of control mechanisms to optimize the configuration of the routers and of the protocols. This architecture interacts with the network equipment and protocols in order to configure the network with the selected protocols and parameters. An analysis of our architecture shows that a real time configuration of routers and a smart selection of the communication protocols bring an important improvement of the performance.

## References

1. Ferber J. *Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence*. Addison Wesley Longman, 1999.
2. Bazzan A.L.C., Wahle J. and Klügl F. Agents in Traffic Modelling - From Reactive to Social Behaviour. KI'99, Bonn, Germany, LNAI 1701, pp 303-307 September 1999.
3. Moukas A., Chandrinou K. and Maes P. Trafficopeter: A Distributed Collection System for Traffic Information. CIA'98, Paris, France, LNAI 1435 pp 34-43, July 1998.
4. Doran J. Agent-Based Modelling of EcoSystems for Sustainable Resource Management. 3<sup>rd</sup> EASSS'01, Prague, Czech Republic, LNAI 2086, pp 383-403, July 2001.
5. Drogoul A., Corbara B. ad Fresneau D. MANTA: New experimental results on the emergence of (artificial) ant societies".in *Artificial Societies: the computer simulation of social life*, Nigel Gilbert & R. Conte (Eds), UCL Press, London, 1995.
6. Bensaid L., Drogoul A., and Bouron T. Agent-Based Interaction Analysis of Consumer Behavior. AAMAS'2002, Bologna, Italy, July 2002.
7. Minar N., Kramer K.H. and Maes P. Cooperating Mobile Agents for Dynamic Network Routing. in "Software Agents for Future Communication Systems", Chapter 12, Springer Verlag, pp 287-304, 1999.
8. Roychoudhuri R., et al. Topology discovery in ad hoc Wireless Networks Using Mobile Agents. MATA'2000, Paris, France. LNAI 1931, pp 1-15. September 2000.
9. Sigel E., et al. Application of Ant Colony Optimization to Adaptive Routing in LEO Telecommunications Satellite Network. *Annals of Telecommunications*, vol.57, no.5-6, pp 520-539, May-June 2002.
10. White T. et al. Distributed Fault Location in Networks using Learning Mobile Agents. PRIMA'99, Kyoto, Japan. LNAI 1733, pp 182-196. December 1999.
11. Bodanese E.L. and Cuthbert L.G. A Multi-Agent Channel Allocation Scheme for Cellular Mobile Networks. ICMAS'2000, USA. IEEE Computer Society press, pp 63-70, July 2000.
12. Wooldridge M. Intelligent Agents. In « Multiagent Systems : a Modern Approach to Distributed Artificial Intelligence » Weiss G. Press, pp 27-77, 1999.
13. Müller J.P and Pischel M. Modelling Reactive Behaviour in Vertically Layered Agent Architecture. ECAI'94, Amsterdam, Netherlands. John Wiley & Sons, pp 709-713, 1994.
14. Pujolle G., Chaouchi H., Gaïti D., Beyond TCP/IP : A Context Aware Architecture, Kluwer Publisher, Net-Con 2004, Palma, Spain, 2004.
15. D. C.Verma, Simplifying Network administration using policy-based management, *IEEE Network* 16(2), 2002.
16. Merghem L., Gaïti D. and Pujolle G. On Using Agents in End to End Adaptive Monitoring. E2EMon Workshop, in conjunction with MMNS'2003, Belfast, Northern Ireland, LNCS 2839, pp 422-435, September 2003.

17. D. Gaïti, and G. Pujolle, Performance management issues in ATM networks: traffic and congestion control, *IEEE/ACM Transactions on Networking*, 4(2), 1996.
18. Gaïti D. and Merghem L.: Network modeling and simulation: a behavioral approach, Smartnet conference, Kluwer Academic Publishers, pp. 19-36, Finland, April 2002.
19. Merghem L. and Gaïti D.: Behavioural Multi-agent simulation of an Active Telecommunication Network, STAIRS 2002, France. IOS Press, pp 217-226, July 2002.