



**HAL**  
open science

## Piloting the spanning tree protocol in home networks using a multi-agent system

Lionel Molinier, Mathieu Ligoeki, Emna Ghedira Molinier, Guy Pujolle,  
Dominique Gaïti

► **To cite this version:**

Lionel Molinier, Mathieu Ligoeki, Emna Ghedira Molinier, Guy Pujolle, Dominique Gaïti. Piloting the spanning tree protocol in home networks using a multi-agent system. *Telecommunication Systems*, 2012, 51 (1), pp.15-27. 10.1007/s11235-010-9411-2. hal-00631616

**HAL Id: hal-00631616**

**<https://hal.science/hal-00631616>**

Submitted on 12 Oct 2013

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Piloting the Spanning Tree Protocol in Home Networks using a Multi-Agent System

Lionel Molinier · Mathieu Ligocki · Emna Ghedira · Guy Pujolle · Dominique Gaïti

the date of receipt and acceptance should be inserted later

**Abstract** Broadband access to the Internet at home was the first step in the emergence of so called Home Networks. In a close future, the number of appliance connected will rise and the network will become the home backbone. Its architecture has to evolve to tackle those new challenges. After a study of the network requirements, this paper introduces a complete system to pilot the forwarding ensuring a proper QoS. This is achieved by a knowledge plane composed of agents embedded on devices, which are optimizing the Ethernet layer.

**Keywords** Home Network · WiFi · HomePlug AV · Agent · QoS · Knowledge Plane · Ontology · Throughput estimation

## Introduction

The spread of broadband accesses and of digital TVs at home is boosting the emergence of so called Home Networks. Currently, Home Networks are only composed of a broadband gateway, a set-top box and the domestic PC. We expect in a close future IPTV, VoIP WiFi

phones, PDA, mobile phones, sensors and even domestic appliances (refrigerator, security devices) to be connected as well.

Contrary to classical networks, the Home Network is not monitored nor configured by an administrator. The end-user is supposed to be a neophyte, without any network skills. This implies strong constraints and the network needs to implement self-\* functionalities, mainly:

- self-configuration: a new device should be easy to connect;
- self-healing: link disruption or device failures must be transparent for the end-user, if possible;
- self-optimizing: the network must reconfigure itself to tackle potential resource shortage.

Those functionalities lead us to autonomic networks techniques, and mainly distributed ones.

In this paper we present a Home Network architecture, widely accepted, in section 1 with a basic routing solutions. However, to improve and implement *self-\** he have used *Distributed Artificial Intelligence* as explained in 2. In 3 we are presenting in details our solution and you are testing it in 4 to complete the work presented in [25, 26]. The multi-agent system involved in those solutions is mostly the same which permits us, in 5, to compare them and also classical ones. Finally, we are concluding this paper by pointing out some work that remain to be achieved in section 6.

## 1 Home Network definition

Major initiatives are working on the Home Network for several years now, in order to identify this emerging network, the use-case, the technologies to be involved and

---

L. Molinier · M. Ligocki · E. Ghedira  
Ginkgo Networks SA, Paris, France,  
E-mail: lionel.molinier@ginkgo-networks.com  
E-mail: mathieu.ligocki@ginkgo-networks.com  
E-mail: emna.ghedira@ginkgo-networks.com

L. Molinier · E. Ghedira · G. Pujolle  
Paris Universit as - LIP6, Paris, France,  
E-mail: lionel.molinier@lip6.fr  
E-mail: emna.ghedira@lip6.fr  
E-mail: guy.pujolle@lip6.fr

D. Gaïti  
University of Technology of Troyes, Troyes, France,  
E-mail: dominique.gaiti@utt.fr

so on. The Home Network cannot be treated separately from its broadband access to the Internet.

### 1.1 Home Network overview

The Broadband Forum is one of the oldest workgroup that mainly focuses on the broadband access. However, since the last years, the Home Network evolution has been taken into account in terms of management. Therefore, the TR-069 [23] jointly with the TR-098 [24] describe and complete and powerful management architecture that tackles both the Home Gateway and others in-home devices. For instance the TR-104 [10] describe a data model for a VoIP end device. However, the Home Network architecture remains out of the scope of the Broadband Forum.

However, this last point is covered by the Home Gateway Initiative (HGI) which is working on defining a wide set of use cases, and on enumerating technical solutions that could be used in a future Home Network. The purpose of this initiative is to mutualize development effort. The main idea is to enable a user to connect all its devices smoothly. For instance, an user must be able to watch a movie stored on the living room DVR directly on its bedroom TV.

Regarding the application, the DLNA [8] (Digital Living Network Alliance) is focusing on the software part of the upcoming use. Devices, DLNA compliant, are able to communicate and exchange content, like in the previously mentioned use case. It is an extension of the UPnP architecture, where the service discovery and management is implemented. Moreover, it defines how audio or video stream should be transported on the network, for instance in terms of codecs.

### 1.2 Architecture

The Home Network architecture is a tricky point to succeed in providing a good user experience in the network. Indeed, even if the software is working well, the network has to carry flows with a good QoS, to be as simple as possible to set-up, and to run standalone, without any network administrator. Moreover this networks must be as cheap as possible since ISP will have to deploy billions.

#### 1.2.1 Today's architecture and evolution

The nowadays architecture is represented figure 1. The Home Gateway is concentrating all services and adapts

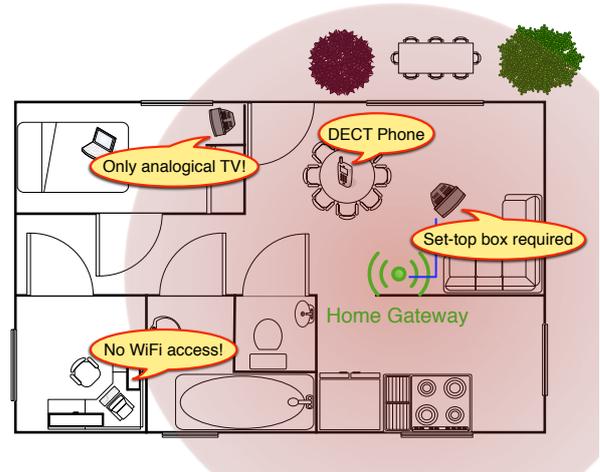


Fig. 1 Nowadays Home Network architecture

the full IP broadband access to each classical technology. For instance the VoIP is converted into PSTN, or eventually DECT.

Obviously, this architecture can only be a transition to a full IP Home Networks. Indeed, in a near future, PSTN phones are to be replaced with IP phones, as for TVs and IP TVs which can be connected to the Home Gateway, using IP. Moreover, the number of devices is expected to rise significantly so that the star topology will no longer be sufficient. Each appliance is intended to be connected to the Home Network, which can be the fridge, the security system, lamps, locks, . . . The Home Network will have to evolve to distributed topologies (mesh or partial mesh), and will have to handle multiple technologies. We are expecting the simultaneous use of WiFi, HomePlug AV, ZigBee, Ethernet, Bluetooth, and any other upcoming technology.

At the same time, the huge bandwidth increase of the broadband access enable users to enjoy 100 Mb/s or even more till the gateway. However, the WiFi alone cannot support such bandwidth, with a wide coverage. Even with the upcoming 802.11n, a single AP cannot cover properly a complete home. A simple and usual way to increase the overall bandwidth of a WiFi network is to add relays and decrease the range of each cell<sup>1</sup>. The following architecture is taking into account this point.

#### 1.2.2 Mesh-hybrid architecture

As discussed in [28], a full mesh architecture is not suitable if the hops number is larger than 2 or 3. We are proposing a 2 layer architecture, introducing new net-

<sup>1</sup> This can be done by refusing the association of farthest clients

work devices, called *bridges*, which have several functionalities, mainly:

- to realize a transparent bridge between technologies (HomePlug AV and WiFi for instance)
- to extend a wireless network (WiFi for instance)

The first level is a full-mesh backbone of those bridging devices. They can connect to each other using Ethernet, PLC, MoCA, or any wireless technology. Connection diversity introduces several paths to reach any other bridge. A smart routing scheme can be used to ensure a proper QoS. Those devices can be seen as wireless extenders.

The second level is for end devices, that can connect directly to any bridge. The connection can be either wireless, or wired. The figure 2 presents a typical instantiation of the architecture.

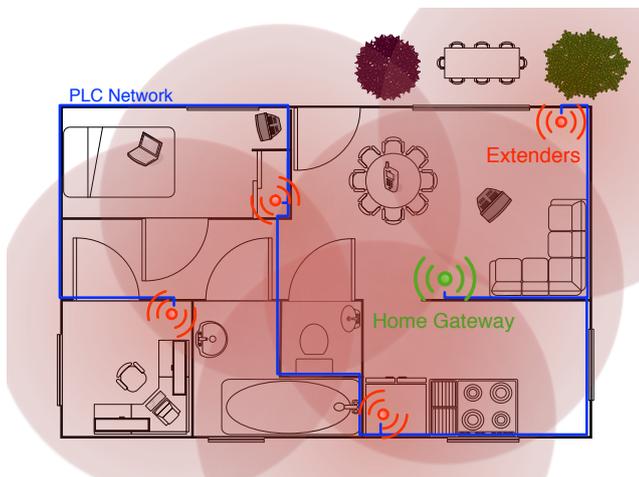


Fig. 2 Home Network architecture

This previous architecture minimizes the modification of end-devices as their are connecting in a standard slave/master architecture (station connected to a WiFi access point). The backbone provides enough diversity to balance load over network resources, and over technologies.

However, the routing between backbone nodes is quite challenging because most of technologies (WiFi and PLC, for instance) are perturbation sensitive, and we have to ensure a perfect QoS to carry HDTV flows, or VoIP flows.

### 1.3 Technologies involved

A Home Network must remain as cheap as possible, so technologies involved are using shared media. The

two key technologies that have been identified so far are WiFi and HomePlug AV. Both are electromagnetic perturbation sensitive and are using the CSMA/CA. Those two points are tricky in terms of QoS.

#### 1.3.1 WiFi

The WiFi is a very well known technology, and well documented in the literature. There were some physical layer models as in [17, 18], some, like [31] are even considering the 802.11e QoS-extension. These methods are widely used in simulated environments. However, to the best of our knowledge, there is no experimentally useful studies to estimate the saturated throughput online with passive methods. In this paper, we introduce a method based on realistic parameters, measured from a publicly available WiFi chipsets (see 4.2).

#### 1.3.2 HomePlug AV

The problem is the same for the HomePlug AV as for the WiFi. The literature about saturated throughput estimation is however less important since the technology is more recent. Nonetheless, we have to mention [14] which develop a model to tackle the CSMA/CA problem with multiple stations on the same medium. However this is not directly useable in our real testbed, that's why we propose, in this paper (see 3.3.3), a method to estimate the saturated throughput. This can be achieved using physical layer statistics that be can retrieved using the open-source Faifa library [1].

### 1.4 Routing concerns

In the section 1.2.2 we have presented a mesh-hybrid architecture and, in 1.3, the shared and perturbation sensitive transmission technology are tricky for the routing scheme. The following is focusing on the routing problematic and the tradeoff between a layer 2 routing or a classical layer 3 one.

#### 1.4.1 Routing problematic

Providing a routing protocol into such networks is not something difficult. There is a lot of routing protocols in the ad-hoc networks literature. The two main are AODV [27] or OLSR [16]. Both are providing connectivity but without any QoS concerns. Some QoS extensions has been proposed, and recently the Hybrid Wireless Mesh Protocol (HWMP) has been standardized in the 802.11s. It introduces the Air Metric in order to introduce QoS in the routing protocol. Unfortunately,

this protocol is only for WiFi ad-hoc networks, and does not encompass our problematic.

Moreover, in the Home Network we have to ensure a good Quality of Experience for the end user, so the HWMP approach is not sufficient anyway. Indeed, the network should not only select the best path available, but must spread out the load over different link so that each flow can enjoy the appropriate QoS.

#### 1.4.2 Layer 2 vs. layer 3

The choice between the layer 2 or the layer 3 is not straight forward. The following paragraphs are discussing this choice.

*Layer 3 experimentation* In previous works [26] we have implemented the Home Network architecture using a layer 3 protocol. Our choice was AODV [27] because of its link to ad-hoc networks. The reactive behavior of AODV fitted well in a wireless network with some mobile devices, and with fluctuating resources.

However, this first try has pointed out some lacks of the layer 3 approach:

- The address scheme is difficult to maintain in a full plug-and-play environment. Each network must be coherent, unique and must provide enough addresses for end-devices. Moreover, the DHCP does not work natively on routed networks.
- Many usual LAN applications such as file sharing are using IP broadcasts, which cannot work with several subnets, which is a natural consequence of routed networks.
- With the proposed architecture, each network should have at least 2 or 3 MAP. Telcos which are to bundle those bridges with their gateways will try to reduce as its minimum the production cost because of the huge number of units. Unfortunately, a layer 3 implementation in bridges implies a faster CPU to decapsulate frames into packets and to encapsulate them back<sup>2</sup>.

*Layer 2 choice* Based on those considerations, we have then examined the layer 2 approach. The proposed architecture is offering diversity, that the layer 2 is not able to use properly. Indeed, diversity means loops in the 802.3 standard. The STP (Spanning Tree Protocol) is running between switches (further called bridges) so

<sup>2</sup> The overhead is situated in the transfer plane of the node, which impacts each packet. The agent presented in section 2 is working in background, in the knowledge plane (which is a user-space application), so that it is scheduled with a lower priority.

that looping links are disabled. This limitation will be discussed further in 3.1.

Apart from this, the ethernet bridging functionality with the STP tackles layer 3 limitations. Indeed, the transfer plane of nodes are much simpler, so that the CPU speed can be reduced.

An other important point is the full independence between Ethernet and IP layers. Indeed, you can change the MAC layer, without impacting the IP one. This separation does not exists in upper layers<sup>3</sup>.

## 2 Agent-based solution

As previously mentioned the complexity of the network architecture in 1.2 and of technologies used cannot be mastered with a classical protocol<sup>4</sup>. Indeed, we have to interface different technologies (with built-in lacks) while guaranteeing QoS. Depending the applicative flow nature, those QoS requirements may differ.

### 2.1 Knowledge plane

Introduced by [15], the knowledge plane can handle this complexity with a reasonable computation overhead. This plane was defined as

[...] a distributed and decentralized construct within the network that gathers, aggregates, and manages information about network behavior and operation.

The knowledge plane enables the network to implement the *self-\** capabilities. Those requirements can be satisfied by a multi-agent system, which provides a decentralized approach to solve problems in complex environments [29]. The followings sections are introducing the agent (see 2.2). In 2.2.2, we are presenting the agent in a society. We are finishing with section 2.3 and 2.4 in which we are describing the platform developed by our company, Ginkgo Networks.

### 2.2 Agent overview

#### 2.2.1 Definitions

According to J. Ferber in [20], the *agent* definition is:

<sup>3</sup> If you change the IP address, you have to restart all your applicative sockets.

<sup>4</sup> The STP is clearly insufficient for this job, but even AODV or OLSR have also shortcomings. To the best of our knowledge, there is no efficient protocol to deal with that.

An agent can be a physical or virtual entity that can act, perceive its environment

However, in our solution, we are using *intelligent agents* which can be defined as:

[an intelligent agent can] communicate with others, is autonomous and has skills to achieve its goals and tendencies. It is in a multi-agent system (MAS) that contains an environment, objects and agents (the agents being the only ones to act), relations between all the entities, a set of operations that can be performed by the entities and the changes of the universe in time and due to these actions.

The figure 3 is presenting a simplified structure of an intelligent agent in the J. Ferber view.

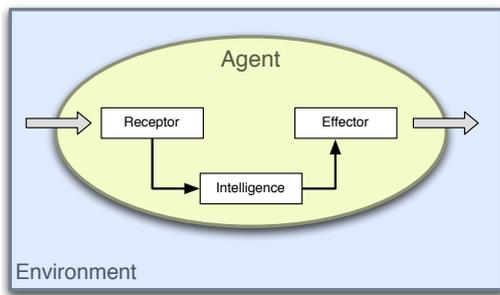


Fig. 3 Simplified structure of intelligent agent

Furthermore, M. Wooldridge in [30] defines them more precisely and emphasizes some characteristics.

Perhaps the most general way in which the term agent is used is to denote a hardware or (more usually) software-based computer system that enjoys the following properties:

- autonomy: agents operate without the direct intervention of humans or others, and have some kind of control over their actions and internal state;
- social ability: agents interact with other agents (and possibly humans) via some kind of agent-communication language;
- reactivity: agents perceive their environment, (which may be the physical world, a user via a graphical user interface, a collection of other agents, the *internet*, or perhaps all of these combined), and respond in a timely fashion to changes that occur in it;
- pro-activeness: agents do not simply act in response to their environment, they are able to exhibit goal-directed behavior by taking the initiative.

### 2.2.2 Agents society

As mentioned in the previous definition, we can exhibit the *collectivity* notion through Multi-Agent Systems. First, we will present the multi-agent environment, the way the agent is perceived, and finally the communication and the interaction between agents.

A multi-agent environment is open<sup>5</sup>, decentralized and composed of a set of *intelligent agents* which are autonomous and distributed. It has to provide protocols for the inter-agent communication and interaction.

Within this environment, an agent is considered as an active object with probing and knowledge representation capabilities. Then, the reasoning ability process them to infer actions.

Interaction, no matter its nature, between agents within a MAS is goal-oriented (either local or global). The coordination is defined as a MAS property executing a specific activity in a common environment. In other words this eases sharing among agents.

Regarding the communication, it enables a better coordination of actions and behaviors to improve the system consistency. The inter-agent communication is based on three levels of understanding:

- syntactic: communication symbol structuring
- semantic: symbol meanings
- pragmatic: symbol interpretation

The inter-agent communication understanding is linked to the semantic (to be understood by other) and to the interpretation (to understand others). Therefore the language used for this communication is derived from the human natural language, with a common and structured vocabulary, called ontology.

After the individual and collective agent overview of the previous section, we will focus now on the Ginkgo MAS.

## 2.3 Ginkgo MAS

An intelligent agent, as defined in section 2.2.1, is embedded on each network device as shown on figure 4. As previously explained, the Ginkgo agent, as an environment entity, has a knowledge representation, a perception of its environment which are both stored in a *situated view*, which is defined in [12].

It is mandatory to have a common vocabulary, both for the inter-agent communication and for the knowledge representation. Therefore an ontology has been developed. The figure 5 is representing a subpart of the ontology using Unified Modeling Language (UML).

<sup>5</sup> An agent can join or leave the community at anytime, without distributing activities

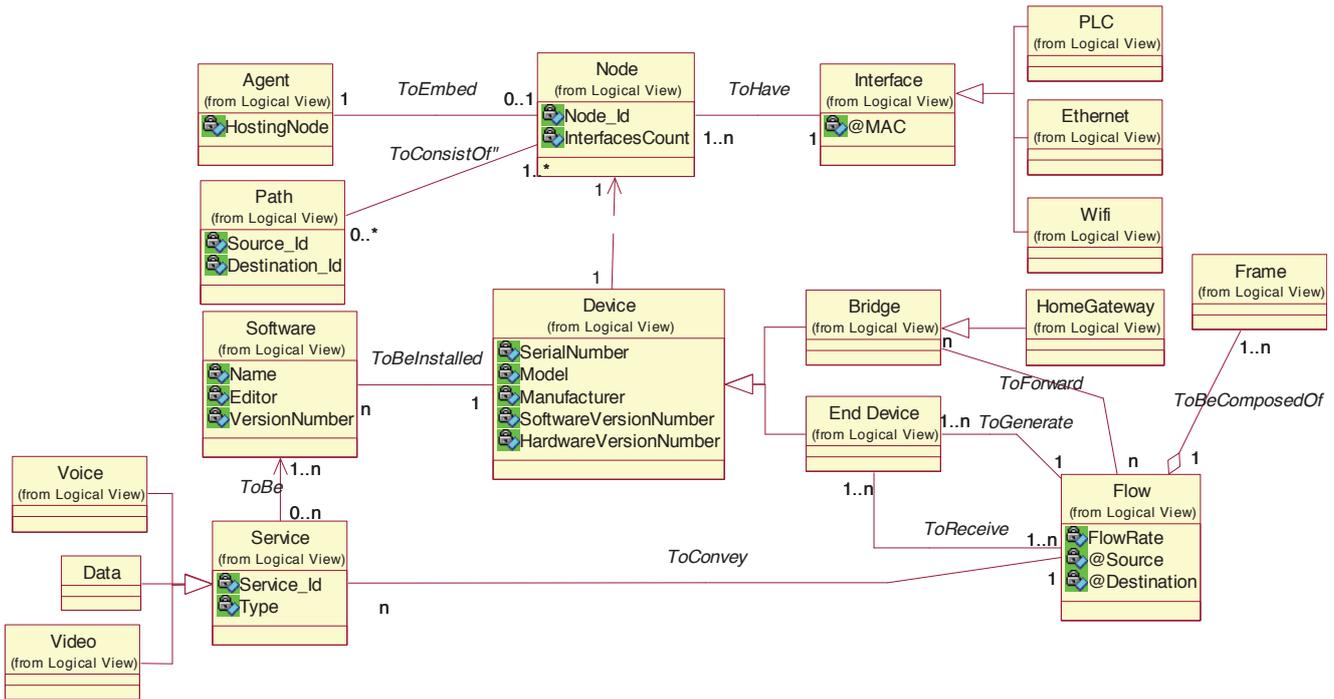


Fig. 5 Extract of the ontology developed

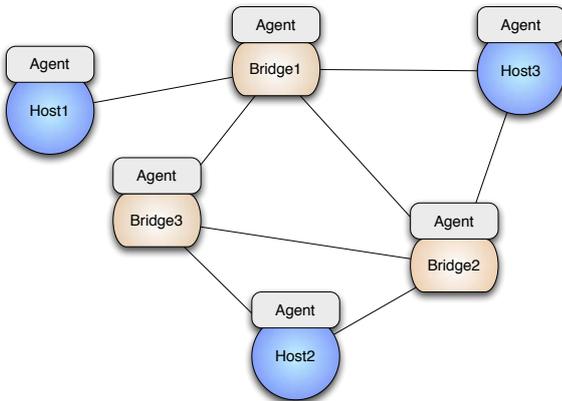


Fig. 4 An agent is embedded each network device

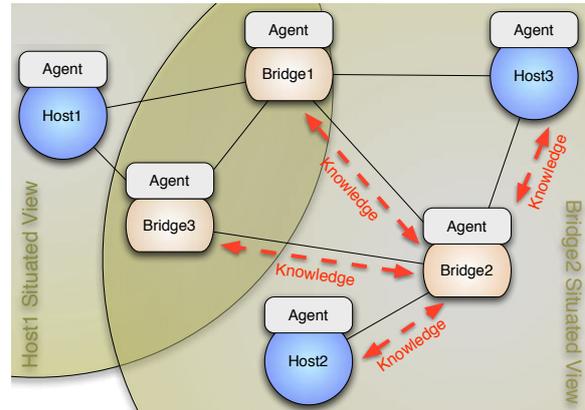


Fig. 6 Agents situated view concept

The inter-agent communication in the Ginkgo MAS is done via the *situated view* exchange. In other words, there is no *speaking act*<sup>6</sup>, but only knowledge exchange. In a MAS, each agent has a partial view of its environment, the Ginkgo MAS has been designed so that an agent can only exchange his *situated view* with its directly connected neighbors (1-hop neighbors), as shown on figure 6.

<sup>6</sup> A *speech act* is a intentional action performed within a communication. There are 5 types of *speaking act*: representative (*it's sunny*), directive (*Open the window*), commissive (*I'll assist to the meeting*), expressive (*I'm happy!*) and declarative (*The court declares the defendant not guilty*)

### 2.4 Agent description

As shown on figure 7, agent capabilities are known as *behaviors* in the Ginkgo agent and are implemented using plug-in module. This set of behaviors is orchestrated by the *dynamic planner* which can start, stop, configure or even modify *behaviors*. The agent's knowledge is stored in the *situated view* which gather both remote and local knowledge. A *behavior* can sense or act on the hosting node using the *effector* and *sensor* interfaces.

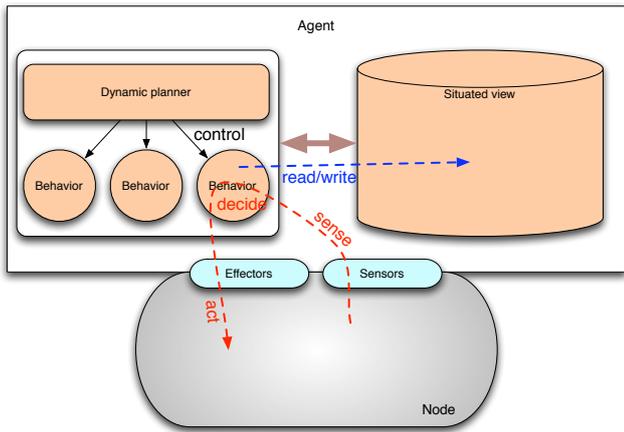


Fig. 7 Agent operation

### 3 Agents piloting MAPs

In a Home Network context, agent’s goal is to change quickly a path when the available resources does not meet flows requirements anymore. For now we have selected 2 different situations:

- link disruption: agents must change the path as soon as a link failure is detected;
- link quality degradation: agents must monitor the link quality, which means the available bandwidth<sup>7</sup>. If a noticeable degradation occurs, agents should change the path.

#### 3.1 STP by-pass

To enable agents piloting the STP, mentioned in 1.4.2, several hooks have been implemented. Indeed, the STP is known to be quite slow to converge after a topology change. In Home Networks context, agents must preempt the normal operation to restore as quickly as possible the applicative flow. This requires a by-pass of the STP during its convergence phase.

In the STP operation, depending its state, a port can forward frames, or block all traffic (except STP PDUs). In our implementation, agents are able to force traffic to pass through blocked ports. This may introduce loops in the network, but internal agents operation prevents this.

In other words, we simply allow agents to modify the forwarding tables of the switch, preventing the STP to override those modifications till its full convergence.

<sup>7</sup> This is an approximation to consider the QoS as a matter of bandwidth, but in most cases, the delay can be converted into extra bandwidth. This works if the delay is considered to be caused only by the serialization problem on the link.

#### 3.2 Building alternatives routes

The figure 8 is describing the agent operation in this precise context of building alternatives routes. Each element of the figure is explained in the following.

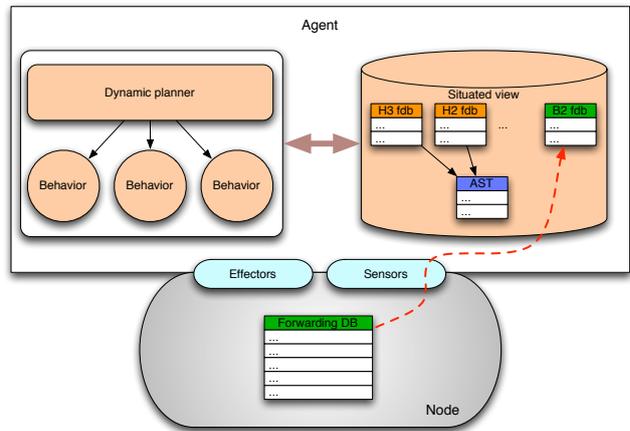


Fig. 8 Agent operation

##### 3.2.1 Local forwarding database

For wireless technologies, the modulation is selected on the fly for each peer. The link performance is defined for the couple (Access Point, Station). To monitor performance, agents have to know to which peer the flow is sent. In other words, they need to know the next hop. This notion does not exist in classical ethernet networks, so agents have to build it back. This can be easily done by comparing the local neighbor list with direct neighbor ones.

##### 3.2.2 Remote forwarding databases

The reactivity is critical for real-time flows. To minimize this time, agents are preparing solutions to any potential problem that may occur. This means, that each agent is looking for an alternate solution for each network access failure<sup>8</sup>.

Actually, each agent is populating the knowledge plane with its own forwarding database where network access identifiers are swapped by the neighbor list reachable with the access in order to identify link without any confusion. Given its neighbor forwarding databases, each agent can compute alternatives.

<sup>8</sup> Actually, it computes a list of alternate solutions, so simultaneous failures could be handled as well.

### 3.3 Link monitoring

Agents have to monitor the link status and performance in realtime, since they have to change as fast as possible the route. These constraints prevent us from using any active measurement methods. Everything must be done locally, in the hosting equipment. Moreover, the computation cost must be taken into account. In a first step of our work, we have elaborated methods as simple as possible<sup>9</sup> to have references for any further works. This section discusses quickly the link status of each interface, and then the maximum available bandwidth estimation on WiFi, and finally on HomePlug AV links.

#### 3.3.1 Link status

Each agent embedded on a MAP must be conscious of network interfaces and must monitor them. Indeed, agent are polling regularly the hardware for information.

#### 3.3.2 WiFi bandwidth estimation

The algorithm used to estimate the WiFi bandwidth is at this step of our work very simple. In digital communication on noisy channel, the physical layer has to optimize the tradeoff between transmission bit rate and noise resilience on the channel. Some algorithms exist to approach this optimality, such as *Atsushi Onoe* [2] algorithm or *SampleRate* described in [11]. We are using this last one in our experiments<sup>10</sup>.

In a word, this algorithm tries each modulation regularly to compute performance metric. In average, it generally uses to best modulation. The tricky point is to test appropriate modulations while not noticeably degrading overall performances. Our estimation is based on those statistics, mainly the average frame transmission time  $tx\_time$  and the time elapse since the last frame send  $last\_tx$ . Those statistics are given for 3 frame sizes, defining the ensemble  $\mathcal{S} = \{250 B, 1.600 B, 3.000 B\}$ . We are introducing a weight function  $w$  to average the frame size:

$$w : s \rightarrow \frac{1}{\sum_{s \in \mathcal{S}} \frac{1}{1+last\_tx(s)}} \frac{1}{1+last\_tx(s)}$$

Finally, the resulting formula is given bellow in the formula 1.

$$wifi\_bw_{max} = \sum_{s \in \mathcal{S}} w(s) \frac{s}{tx\_time(s)} \quad (1)$$

<sup>9</sup> The strongest approximation is the CSMA/CA effect on shared mediums.

<sup>10</sup> There is no strict limitation in using *SampleRate*, since most of the rate selection algorithm are using the same principle.

In practice, the algorithm is working with 2 steps: first, a geometric averaging of the formula 1 is computed to smooth high frequency variation. Secondly, given the first order derivate, the next value is extrapolated. This algorithm is very simple and must be consider a lower bound of the estimation efficiency. As discussed in 6.2, a more powerful algorithm is in development.

#### 3.3.3 HomePlug AV bandwidth estimation

As for the WiFi, the HomePlug AV physical layer as to adapt to the channel conditions by adjusting the modulation. However, despite the WiFi, there are 1152 frequencies available for the communication (see [22] for more information). The resulting modulation table is called the Tone Map, from which we can compute the channel capacity, called  $cap_c$ .

The HomePlug AV chip is also maintaining statistics on successfully transmitted frames<sup>11</sup> and unsuccessfully ones. We are calling  $p_{suc}$  the proportion of successfully transmitted frames.

Since there is a high binary error rate on power lines, the HomePlug AV is using a powerful FEC<sup>12</sup> system based on Turbo Codes. This adds overhead in blocks, so that the payload bandwidth is reduced by a factor  $p_{FEC}$ . As explained in [21], the factor value is  $p_{FEC} = \frac{1}{2}$ .

Finally, the bandwidth can be computed by the formula 2:

$$hpav\_bw_{max} = p_{suc} \cdot p_{FEC} \cdot cap_c \quad (2)$$

In practice, we are smoothing this value so that high frequency variation are filtered out. One more time, this is the simplest way to estimate the bandwidth, that is sufficient in simple environment. A better estimation is under development, as discussed in 6.2.

## 4 experimental set-up

To experiment previous concepts, we have decided to setup a real testbed instead of using a simulator because many potential problems can be caused by implementation shortcuts. We have simplified the architecture to only 3 PC running GNU/Linux, in order to isolate problems quickly. The figure 9 is illustrating the testbed. We focus on Ethernet (for the link disruption scenario) and on WiFi or HomePlug AV (for link quality degradation scenario).

<sup>11</sup> In HomePlug AV, we prefer talking of blocks

<sup>12</sup> FEC: Forward Error Correction

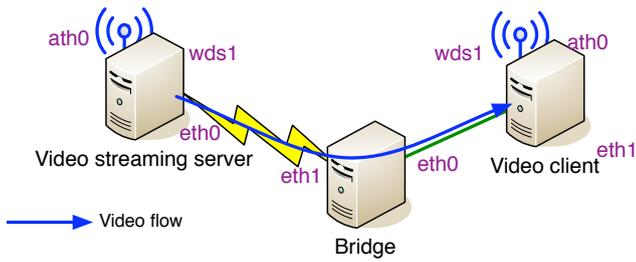


Fig. 9 Testbed network

#### 4.1 Bridge implementation

The GNU/Linux kernel is implementing the ANSI/IEEE 802.1d [3] standard. We have patched the STP to allow our agent by-passing ports states as described in 3.1.

#### 4.2 WiFi infrastructure

The classical infrastructure mode of WiFi cannot be used in conjunction with ethernet bridging. Indeed, the 802.11 standard is using the MAC address to identify wifi station (association, modulation selection, ...), so that they cannot forward foreign MAC addresses. The WDS introduces a 4 addresses header in 802.11 frames to enable frame forwarding.

We have used this system in conjunction with Atheros [4] based WiFi cards. The Madwifi [2] module for GNU/Linux is supporting the WDS, and has been patched for performance monitoring.

#### 4.3 HomePlug AV infrastructure

For our HomePlug AV network, we have used Develo dLAN 200 ethernet adapters [5]. There are running with Intellon [6] chipset, like mostly any other adapters. Thanks the Faifa [1] library we were able to retrieve layer 1 statistics in our system. For instance, we have access to the current Tone Map used to reach a given destination.

As a first step in our experimentation we have not studied the effect of concurrent HomePlug AV stations. The PLC network in figure 9 is a real power line, with only our 2 adapters connected. This prevent us from consider the CSMA/CA effect on the QoS.

## 5 Results

Given the network architecture presented in 4, we have realized two sets of experimentations. The first one was

performed to check agents ability on detecting and recovering from a link disruption. The second one to measure agents reactivity when faced to a WiFi perturbation, then faced to a HomePlug AV one.

#### 5.1 Ethernet wire unplugging

Another part of our experimentation was about wire failures. Indeed, when an agent detects a link down it should find an alternative access. Based on the previous scenario, the flow is going from the *Video streaming server* to the *Video client* through the *Bridge*. The link between the *Video server* and the *Bridge* is then unplugged.

		Recovery time
Layer 3	with Agents	5.4 s
	OSPF only	11 s
Layer 2	with Agents	680 ms
	STP only	21 s
	RSTP only	$\approx 1$ s

Table 1 Recovery time after a link failure

Values given in table 1 have been measured using a network packet generator [7] with a periodical packet generation. Both OSPF and STP implementation are running with default configuration values.

In this experimentation, the tricky point is the ability to act on the network device remaining transparent for applicative flows. There is a huge difference between the layer 3 approach. Indeed, the IP stack does not allow modification of interface state without a long disruption. This point is blocking because it prevents agent from acting properly. Fortunately, the separation between the layer 2 and the layer 3 is good enough to allow agent to operate.

We have measured, with the layer 2 approach an average disruption of 540 ms (see table 1). This delay is mainly cause by slow software. Indeed, up to now, agents are developed in Java to ease the development and tests<sup>13</sup>. An optimized software design can reduce it to less than 50 ms.

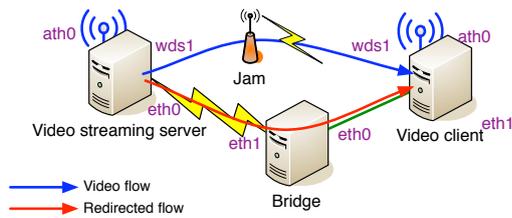


Fig. 10 WiFi perturbation scenario

## 5.2 WiFi perturbation

Like explained in 3.3, agents are monitoring WiFi performances to ensure that flows can be forwarded in good conditions. If a problem is detected, the previously computed alternative list helps agent to change flow path. In the setup, a flow is going from the *Video streaming server* to the *Video client* by the WDS link. An agent is running on each node, so that they can fill their situated view to compute alternatives.

As shown on figure 10, we then generate a perturbation on the WDS link frequency, so that MAC layer performances collapse. As a consequence, the agent situated on the *Video server* notices the problem and solves it.

Figure 11 is showing that agents are reacting to the perturbation in less than 3 seconds. Even if this is not satisfying for real time flows, this time can easily be reduced as discussed below. Anyway, this detection time is tricky to minimize applicative perturbations. No matter we are using the layer 2 approach or the layer 3, the reactivity time is mostly the same.

However in our testbed, the generated perturbation is the worst case that may occur in real field because resources are collapsing in few milliseconds as shown on figure 12. When the agent detects the perturbation, the applicative flow is already degraded. A real perturbation should be smoother, in most cases.

## 5.3 HomePlug AV perturbation

Figure 13 illustrate the perturbation scenario for the HomePlug AV. Results are mostly the same as for WiFi, since in both case, the overall performance relies on the perturbation detection. However, figure 14 is showing the performance of our bandwidth estimation in the test architecture. The section 6.2 will describe work that remains to be done to take into account the CSMA/CA effect.

<sup>13</sup> Our agent framework can be executed indifferently on an own-made simulator or on GNU/Linux. This causes an important overhead in abstraction layer that considerably slows agents execution.

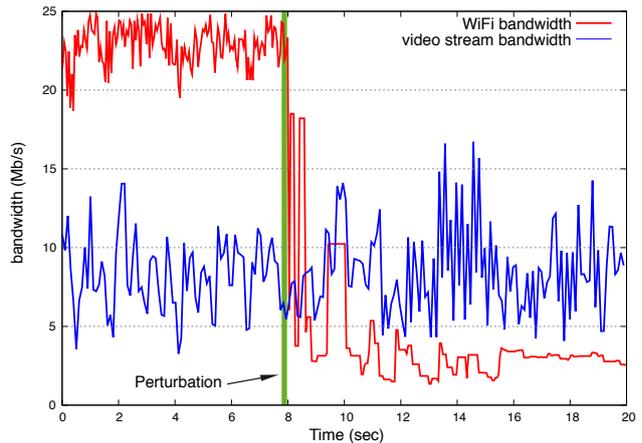


Fig. 12 WiFi perturbation profile

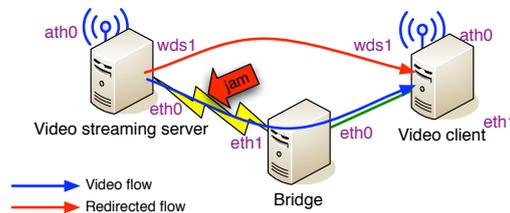


Fig. 13 HomePlug AV perturbation scenario

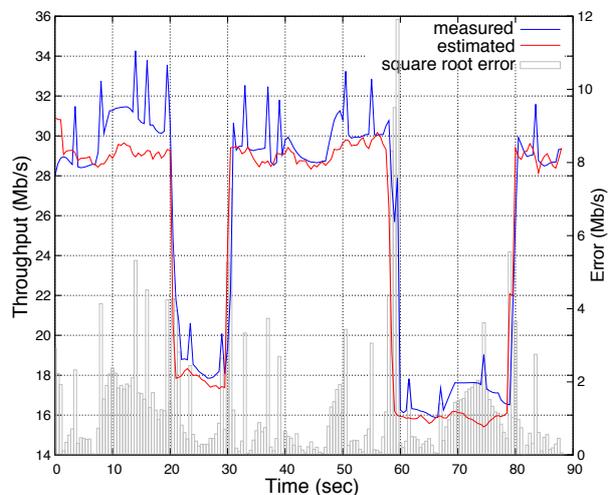
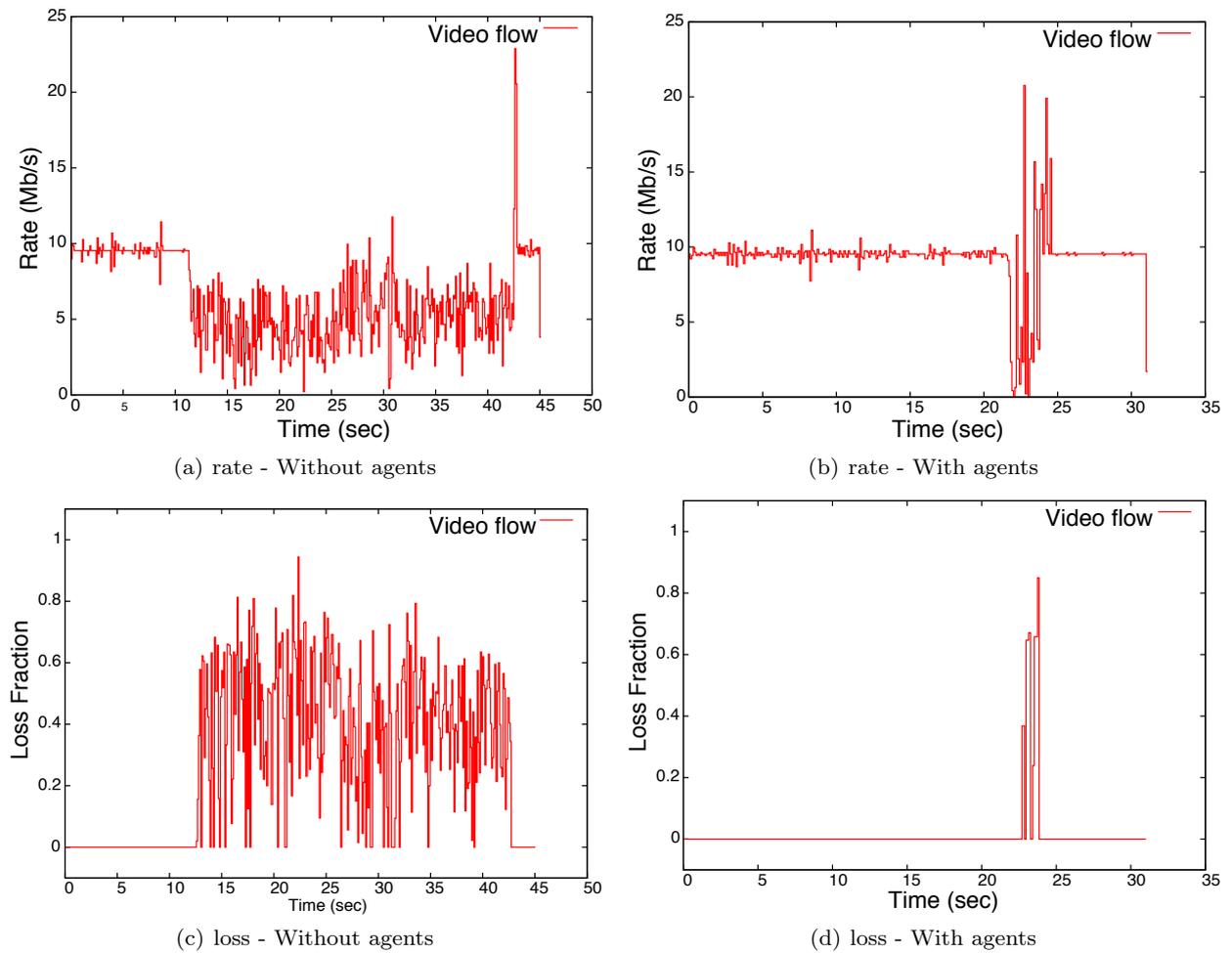


Fig. 14 HomePlug AV bandwidth estimation

## 6 Future works

In previous sections, we have presented the third step of our work. The first one was to simulate the knowledge plane on an own-made simulator to validate agents operations. As a second step we have setup a real testbed using a layer 3 routing protocol. Then in this third step, we have used the same testbed with a layer 2 approach. However a lot of work remains to be done.



**Fig. 11** Rate and loss factor during a perturbation.

### 6.1 Architecture extension

The architecture used for these experimentations is not yet complete, and one or two more PC will be added soon. This will enable more complex scenario, where agents will have to cooperate more intensively.

### 6.2 QoS measurements

The passive QoS measurements both for WiFi or Home-Plug AV is still under study to improve the confidence in the estimator in one hand, and in the other hand to handle the CSMA/CA effect. Some AI techniques like kNN or neuronal networks are tested. At the same time the agent framework is used to simulate the CSMA/CA behavior.

Another way of development is to introduce a prediction module with the bandwidth temporal serie so that the agent can change the path faster, and ideally

before the perturbation became damaging for the user experience.

### 6.3 Knowledge plane

Another way of investigation will be to improve the knowledge plane. Indeed, we are using a very simple technique which relies on forwarding tables exchange. However, this requires a stable network in which forwarding tables can be considered as stable (at 10 seconds scale).

In highly variable environment, such method may lead to inconsistency which are unrecoverable by layer 2 mechanism. Methods using the ant-routing principle, presented in [13, 19], can be used successfully since the network diameter is small.

## 6.4 Handover management

For the WiFi, the WDS allows horizontal handovers but it has been eluded. Next work will be to integrate an algorithm to optimize the handover decision based on several criteria, such as the number of stations connected, the required bandwidth. . . The paper [9] presents the handover work on the same SMA platform that can be integrated with this work.

## 6.5 Service differentiation

In the presented architecture, we are working with only one flow (IPTV). In the real field Home Network, there will be more flows, such as Data, VoIP, Game, IPTV, . . . We are working on isolating flows on separated ethernet VLAN, so that the agent system can route services independently. Moreover, a *DiffServ*-like architecture is implemented in each forwarding device. This enables also the SMA to tune the QoS parameters, so that an agent can reduce bandwidth allocated for the data, in order to preserve the VoIP quality.

## 7 Conclusion

The results presented in section 5 are confirming that the layer 2 approach is more effective than the layer 3 (see 1.4.2). The independence between the Ethernet layer and the IP one enables agents (and any piloting system) to work without impacting the end user experience.

The work on this layer is harder, but agents cross-layer capabilities helps us to transpose upper layer notions without modifying the transfer plane of network nodes. This transfer plane remains cheap and efficient.

Home Networks are using perturbation sensitive technologies due to their low deployment costs. However, users are requiring high quality of service as this network is the backbone of tomorrow homes. Dealing with this antagonism is not as simple as elaborating a new protocol. It as been experimented in 5 that a multi-agent system is able to master such complex environments.

The first real experimentations are quite satisfying as they can solve partially many problems. The real field is still far, but we think that progressing step by step, adding new components at each step may lead to a complete working system fitting all requirements.

## Acknowledgment

The authors would like to thank the Ginkgo Networks developer team that helps them in implementing this solution, and one of our closest CELTIC/AutHoNe partner, Orange Labs for their guidelines and expertise on such networks.

## References

1. URL <https://dev.open-plc.org/>
2. URL <http://madwifi.org>
3. URL <http://www.linuxfoundation.org/en/Net:Bridge>
4. URL <http://www.atheros.com>
5. URL <http://www.devolo.com>
6. URL <http://www.intellon.com/>
7. URL <http://cs.itd.nrl.navy.mil/work/mgen/>
8. Dlna overview and vision whitepaper. Tech. rep., Digital Living Network Alliance (2007). URL <http://www.dlna.org>
9. Abid, M., Ligocki, M., Molinier, L., Nguenguang, G., Pujolle, G., Gaiti, D., Zimmermann, H.: Practical handover optimization solution using autonomic agent-based piloting system. pp. 1–5 (2008). DOI 10.1109/WD.2008.4812921
10. Bernstein, J., Stark, B.: Dslhome provisioning parameters for voip cpe. Tech. rep., Broadband Forum (2005)
11. Bicket, J.: Bit-rate selection in wireless networks. Ph.D. thesis, Massachusetts Institute of Technology (2005)
12. Bulot, T., Khatoun, R., Hugues, L., Gaiti, D., Merghem-Boulahia, L.: A situatedness-based knowledge plane for autonomic networking. *Int. J. Netw. Manag.* **18**(2), 171–193 (2008). DOI <http://dx.doi.org/10.1002/nem.679>
13. Caro, G.D., Dorigo, M.: Antnet: A mobile agents approach to adaptive routing. Tech. Rep. 12, Universite Libre de Bruxelles, IRIDIA (1997). URL [citeseer.nj.nec.com/dicaro97antnet.html](http://citeseer.nj.nec.com/dicaro97antnet.html)
14. Chung, M.Y., Jung, M.H., Lee, T.J., Lee, Y.: Performance analysis of homeplug 1.0 mac with csma/ca. *Selected Areas in Communications, IEEE Journal on* **24**(7), 1411–1420 (2006). DOI 10.1109/JSAC.2006.874404
15. Clark, D.D., Partridge, C., Ramming, C.J., Wroclawski, J.T.: A knowledge plane for the internet. In: SIGCOMM '03: Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications, pp. 3–10. ACM Press, New York, NY, USA (2003). DOI 10.1145/863955.863957. URL <http://portal.acm.org/citation.cfm?id=863957>
16. Clausen, T., Jacquet, P.: Optimized link state routing protocol (olsr). RFC 3626 (Experimental) (2003). URL <http://www.ietf.org/rfc/rfc3626.txt>
17. Dao, N., Malaney, R.: Throughput performance of saturated 802.11g networks. pp. 31–31 (2007). DOI 10.1109/AUSWIRELESS.2007.82
18. Dao, N., Malaney, R.: A new markov model for non-saturated 802.11 networks. pp. 420–424 (2008). DOI 10.1109/ccnc08.2007.100
19. Dhillon, S.S., Mieghem, P.V.: Performance analysis of the antnet algorithm. *Comput. Netw.* **51**(8), 2104–2125 (2007). DOI <http://dx.doi.org/10.1016/j.comnet.2006.11.002>
20. Ferber, J.: Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence. Addison-Wesley Professional (1999)

21. Guerrieri, L., Bisaglia, P., Dell'Amico, G., Guerrini, E.: Performance of the turbo coded homeplug av system over power-line channels. pp. 138–143 (2007). DOI 10.1109/ISPLC.2007.371112
22. Hazen, M.E.: The technology behind homeplug av powerline communications. *Computer* **41**(6), 90–92 (2008). DOI <http://doi.ieeecomputersociety.org/10.1109/MC.2008.205>
23. Lupton, W., Blackford, J., Digdon, M., Spets, T.: Cpe wan management protocol - amendment 2. Tech. rep., Broadband Forum (2007). URL <http://www.broadband-forum.org/technical/download/TR-069Amendment2.pdf>
24. Lupton, W., Bouchat, C.: Internet gateway device data model for tr-069 - amendment 2. Tech. rep., Broadband Forum (2008). URL [http://www.broadband-forum.org/technical/download/TR-98\\_Amendment\\_2.pdf](http://www.broadband-forum.org/technical/download/TR-98_Amendment_2.pdf)
25. Molinier, L., Ligocki, M., Pujolle, G., Gaiti, D.: Pilot-ing the spanning tree protocol in home networks using a multi-agent system. pp. 1–5 (2008). DOI 10.1109/WD.2008.4812910
26. Nguengang, G., Molinier, L., Boite, J., Gaïti, D., Pujolle, G.: Intelligent routing scheme in home networks. In: Springer (ed.) *Home Networking*, pp. 179–196 (2008)
27. Perkins, C., Belding-Royer, E., Das, S.: Ad hoc on-demand distance vector (aodv) routing. RFC 3561 (Experimental) (2003). URL <http://www.ietf.org/rfc/rfc3561.txt>
28. Shihab, E., Cai, L., Wan, F., Gulliver, A., Tin, N.: Wireless mesh networks for in-home iptv distribution. *Network, IEEE* **22**(1), 52–57 (2008). DOI 10.1109/MNET.2008.4435903
29. Wooldridge, M.: *An Introduction to Multi-Agent Systems*. John Wiley & Sons, Inc., New York, NY, USA (2001)
30. Wooldridge, M., Jennings, N.R.: Intelligent agents: Theory and practice. *Knowledge Engineering Review* **10**(2), 115–152 (1995). URL <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.55.2702>
31. Xiong, L., Mao, G.: Saturated throughput analysis of ieee 802.11e using two-dimensional markov chain model. In: *QShine '06: Proceedings of the 3rd international conference on Quality of service in heterogeneous wired/wireless networks*, p. 10. ACM, New York, NY, USA (2006). DOI <http://doi.acm.org/10.1145/1185373.1185387>