THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par *l'Université Toulouse III - Paul Sabatier*
Discipline ou spécialité : *Réseaux, Télécommunications, Système et Architecture*

Présentée et soutenue par *Lucile Canourgues*
Le *20 Mai 2008*

Titre :
Algorithmes de Routage dans les Réseaux Mobile Ad hoc Tactique à Grande Echelle

---

JURY
Guy Pujolle, Président du Jury, Professeur, Université, Paris VI
André-Luc Beylot, Directeur de thèse, Professeur, INPT/ENSEEIHT
Philippe Jacquet, Rapporteur, Directeur de recherche, INRIA Rocquencourt
Khaldoun Al Agha, Rapporteur, Professeur, Université Paris XI
Fabrice Valois, Examinateur, Maître de Conférence HDR, INSA Lyon
Jérôme Lephay, Examinateur, Ingénieur, Rockwell Collins France

---

Ecole doctorale : *Mathématiques,Informatique et Télécommunications de Toulouse*
Unité de recherche : *Institut de Recherche en Informatique de Toulouse (IRIT)*
Directeur(s) de Thèse : *André-Luc Beylot,Professeur, INPT/ENSEEIHT*
Rapporteurs : *Philippe Jacquet, Directeur de recherche, INRIA Rocquencourt*
Khaldoun Al Agha,Professeur, Université Paris XI

Remerciements

Pour la plupart des doctorants, la page de remerciement est celle qu'on écrit en dernier. On repousse toujours cette rédaction de peur d'oublier quelqu'un, peut être aussi pour prolonger encore un peu ces trois années de travail: on sait qu'une fois cette dernière page apportée au manuscrit, elle clôturera une fois pour toute le chapitre estudiantin de notre vie. Cependant, il faut bien se lancer et accepter cette fin qui n'est en fait que le début d'une autre vie. Je m'excuse par avance auprès des personnes que je pourrais oublier et j'espère qu'elles ne me tiendront pas rigueur de cet oubli involontaire.

Je tiens tout d'abord et en premier lieu à témoigner ma gratitude à mes deux encadrants qui m'ont soutenu et orienté tout au long de cette thèse. On pourrait comparer une thèse CIFRE à un tabouret à 3 pieds où l'encadrant académique, l'encadrant industriel et le doctorant servent à part égale à la réussite et à la stabilité de la structure. Mes deux co-encadrants ont été véritablement les deux socles sur lesquels j'ai pu me reposer. Merci à André-Luc Beylot, professeur à l'Institut National Polytechnique, mon directeur de thèse, pour m'avoir encadré lors de mes études d'ingénieur au cours desquelles il a pu me faire découvrir le domaine des réseaux, puis pour m'avoir fait confiance en me confiant ce sujet de thèse et pour m'avoir accompagné, encouragé et conseillé au cours de ces 3 ans tant sur le plan technique que humain. Merci à Jérôme Lephay, Ingénieur Système Senior au sein de la société Rockwell Collins France qui m'a accueilli, orienté, encadré et soutenu durant 3 ans, facilitant mon intégration au sein de la société et veillant continuellement à ce que je puisse m'épanouir dans mon travail. Plus que le co-encadrant de cette thèse, il a été un exemple, une oreille toujours à l'écoute, un soutien moral lors des conférences et un partenaire de brainstorming technique ou de discussions diverses et culturelles.

Je souhaiterais ensuite remercier les deux relecteurs de ma thèse, Khaldoun Al Agha et Philippe Jacquet pour avoir accepté de relire mon travail et pour leurs commentaires plus que constructifs qui m'ont permis d'améliorer mon manuscrit. Je voudrais aussi remercier Guy Pujolle pour avoir accepté d'être le président du mon jury malgré un emploi du temps que tout le monde sait tres chargé.

Je remercierais ensuite mes collègues de Rockwell Collins France, membre de l'équipe ISDR et plus généralement du service Système de Communications Militaires. Je pense tout particulièrement à Monique Escalette, Chef de Service Système de Communications Militaires, à Laurent Soyer, Chef de Projets Radio

Logicielle et à Bernard Bouillaud, Responsable de Programme Radio Logicielle. Tous les 3 ont facilité mon intégration au sein de leurs équipes. Je les remercie pour la confiance et le soutien qu'ils m'ont témoigné ainsi que pour les excellentes conditions dans lesquelles ils m'ont permis d'effectuer mon travail de recherche.

Je n'oublierais bien évidemment pas tous les membres du laboratoire TeSA, à commencer par sa plus emblématique représentante, Marie-Josée Estepa. Je m'attarderais un instant pour saluer les membres du bien nommé "bureau de la culture", Ferdinand, Patrice, Julien, Mariana et le p'tit Manu. Il me sera difficile d'oublier nos nombreuses discussions ou débats tant sémantiques que historiques et parfois techniques. Je n'aurais jamais tant appris sur la grammaire française, sur les contrepétries et sur la vie de Napoléon que durant ces 3 années.

Arrivée à ce point de ces remerciement, je désire me retourner vers l'ensemble de mes ami(e)s qui sont un soutien au quotidien. Je remercierais plus particulièrement "les filles" qui, plus que des amies, sont la deuxième famille que j'ai choisie, qui est toujours présente, bienveillante, encourageante et attentionnée dans tous les moments de ma vie.

Merci, cela va sans dire, à mes parents pour m'avoir soutenu, encouragé et accompagné depuis toujours.

Merci à Florent, mon frère, qui a toujours été pour moi un modèle, qui m'a permis de me dépasser et a si souvent su me redonner espoir.

Merci à Gael, mon compagnon, qui n'a cessé de m'épauler, de m'encourager dans la voie que j'ai choisie et a su tout faire pour que jamais je ne renonce, ni ne perde confiance. Merci de croire autant en moi.

Contents

List of Figures

List of Tables

Abstract

The current Transformation of the military networks adopts the MANET as a main component of the tactical domain. Indeed, a MANET is the right solution to enable highly mobile, highly reactive and quickly deployable tactical networks. Many applications such as the Situational Awareness rely on group communications, underlying the need for a multicast service within the tactical environment where the MANET is employed as a transit network. The purpose of this thesis is to study the setting up of an optimal multicast service within this tactical environment. We firstly focus on defining the protocol architecture to carry out within the tactical network paying particular attention to the MANET. This network is interconnected with different types of networks based on IP technologies and implementing potentially heterogeneous multicast protocols. The tactical MANET is supposed to be made of several hundred of mobile nodes, which implies that the scalability is crucial in the multicast protocol architecture choice. Since the concept of clustering proposes interesting scalability features, we consider that the MANET is a clustered network. Thereby, we define two multicast routing protocols adapted to the MANET: firstly STAMP that is in charge of the multicast communications within each cluster and secondly SAFIR that handles multicast flows between the clusters. These two protocols that can be implemented independently, act in concert to provide an efficient and scalable multicast service for the tactical MANET. Then, we study the interoperability of these multicast protocols employed within the MANET with those employed in the heterogeneous networks that it is interconnected with in order to guarantee end-to-end seamless multicast services to users. Finally, since the multicast protocols proposed in this thesis rely on underlying unicast routing protocols, we propose, in the last chapter, a scalable unicast routing protocol based on OLSR.

Chapter 1

Introduction

For the last twenty years, the emergence of new technologies of information and communication has radically modified society. Inspired by this revolution that has led us to what is called the "Information Age", the military vision operates a Transformation, evolving from a platform-centric force to a network-centric force. The concept of network-centric operation stands the information in the sense of strategic, operational and tactical intelligence in the first row of all DoD concerns.

In the first part the concept of Network Centric Warfare (NCW) and its implementation through the Global Information Grid (GIG) are presented since they represent the future of the warfare. The Global Information Grid imposes some modifications to the traditional military architectures such as the tactical communication architecture. Thus, the new architecture of the tactical communications is presented in the second section. As a central part of the tactical communications, the tactical Mobile Ad hoc Network (MANET) and its specificities are described and analyzed in the third part. After this third part, the context of our work will be fully defined. Therefore, the objectives of the thesis and finally the organization of the thesis will be presented.

1.1 The future of the warfare

1.1.1 The Network-Centric Warfare concept

The concept of Network Centric Warfare, also called Network-Enabled Capabilities (NEC) by the British or Network-Based Defense (NBD) by the Swedish, emerged in 1998 in the United States as a new way of thinking about military operations in the Information Age. The US military wanted to use in their own area the information technologies and more particularly the network communication technologies that were developing in the civil area [116]. Several definitions, more or less concrete, of the NCW concept may be found. The NCW is defined by the US military as the combination of emerging tactics, techniques and procedures that a networked force may use to obtain a decisive military advantage [17]. In a less abstract way, this concept can also be defined as the use of communication and information systems to create a shared knowledge of the battlefield in order to allow a more efficient control of the deployed military actors, a better decision making of the command in the field and a shorten of the sensor to shooter loops. Finally, in a simplified way, the NCW concept may be defined as the networking of command, control and weapon systems i.e. all the military actors thanks to the New Technologies of Information and Communications (NTIC) [29, 87]. Through the NCW vision, the hierarchical approach of the military communications is abandoned in favor of a more reactive and efficient communication architecture. Indeed, the network connects everyone permanently, and if possible in real time, from the commander to the warfighter on the field. All the theaters of operations (ground, maritime, airborne and satellite) are concerned. The battlefield is no longer a simple hierarchical network but a network of networks, "a global grid of multiple, interoperable, overlapping sensor, engagement and command nets". In the US definition, this network of networks is named the Global Information Grid (GIG).

1.1.2 The Global Information Grid

The GIG is a globally connected, end-to-end set of information capabilities, associated processes, and personnel for collecting, processing, storing, disseminating, and managing information on demand. GIG users (people, processors, sensors, etc.) are either producers of information or consumers of information. Producers "publish" their information to the shared information space provided by the network, and consumers can access to this data by searching and retrieving, or subscribing to the data. All users can get the relevant information they need at any time. As new data is published, its availability becomes known to those subscribers needing that information, thereby enriching their situational awareness. At the heart of the GIG is an Internet-like communication network that provides the underlying connectivity among the users. The most important attribute of the GIG is the rich, robust connectivity it will provide to anybody connected to the common network [105].

1.2 A new architecture for the tactical communications

1.2.1 Current tactical communication architecture

Traditionally, tactical communications (communications occurring in the tactical operation region, the operational battlefield) were supported by a hierarchical architecture made of Trunk communication subsystems that interconnect headquarters at the brigade level and above, of Combat Net Radios (CNRs) that provide communications support to combat troops at the brigade level and below, and of Local Area Subsystems for communications within headquarters or between different vehicles At The Halt. Within this architecture illustrated by the figure 1.1, existing legacy communication systems (and applications) are tailored for very specific needs. With the existing legacy systems, the network is static and well defined; communications are point-to-point; critical communications nodes are predetermined and operate At The Halt, remaining in a fixed location during the transmission.

1.2.2 The Tactical Internet

The concept of Network Centric Warfare and its implementation through the GIG suppose large scale communication systems capable of seamless connectivity across the traditional boundaries guarantying data transmission with deterministic delay. The future communication network must be able to support Situational Awareness (SA) to commanders in nearly real-time to provide an accurate knowledge of the situation on the battlefield. The hierarchical communication schemes are not well suited for this purpose. Moreover, commanders must be able to control troops regardless of their position. It is therefore necessary to maintain communications such as mailing, data diffusion and voice, continuity from deployment beginning at the barracks out to the battlefield even On The Move. This capability is not supported by the legacy systems. Consequently, to support the transformation of the warfare vision and the requirements for network communications it infers, the Tactical Net-

work, also called the Tactical Internet is defined to replace the traditional tactical communication systems.

The Tactical Internet is made of several parts described hereafter.

- The Wide Area System (WAS) interconnecting the headquarters and the main Control Points.
- The Mobile Tactical Internet made of Tactical Communication Nodes (TCN) which provides new radio waveform capabilities to interconnect combat troops. The Mobile Tactical Internet is connected to the WAS through point to point or point to multipoint high capacity links. This system provides transversal connectivity to the battalion, company and platoon levels.
- The Local Area Subsystem that remains the same as in the current tactical communication scheme.

The principal challenge of this new architecture illustrated by the figure 1.2 is the Mobile Tactical Internet. Indeed, it must provide efficient communications to the combat troops which are highly dynamic. The Tactical Internet is composed of a variety of heterogeneous transmission networks such as LANs, satellites networks, legacy CNR and commercial networks. It is the part of the Tactical Communication Nodes to interconnect all these networks. A TCN is a vehicular platform that integrates radio equipments and networking functions, allowing to set up a seamless IP network over heterogeneous radio networks. It may include radio communication equipments, LANs, routers, hosts, gateways, servers or management workstations. The Tactical Communications Nodes form together a radio network that is highly dynamic (i.e. the number and the type of nodes would change uncertainly), self-configuring, made of highly mobile nodes and should operate On The Move. The TCN network should be deployable anywhere, anytime providing communications continuity even when nodes move. The ideal candidate solution for this fully mobile and dynamic tactical communication network appears to be the Mobile Ad hoc Networks (MANET) concept [84]. In the remainder of the document, the radio communication equipment of a Tactical Communications Node will be referred to as a MANET node, and the radio network made by these radio equipments will be referred to as the Tactical MANET.

1.3 The tactical Mobile Ad hoc Network

1.3.1 Definition of a MANET

MANETs found their origins in the early 1970s in a program sponsored by the DARPA [31] for military applications, the “packet radio network” (PRNET) project [67]. The DARPA wanted a wireless packet network for military use. This wireless system was expected to be distributed, multi-hop, self-organizing and self-configuring. Then, this project evolved to the survivable radio network (SURAN) project to improve scalability and survivability, which is the ability of the network to continue operating even when the network or a link fails. During the 1980s, researches on the area of ad hoc network in military environment were extremely

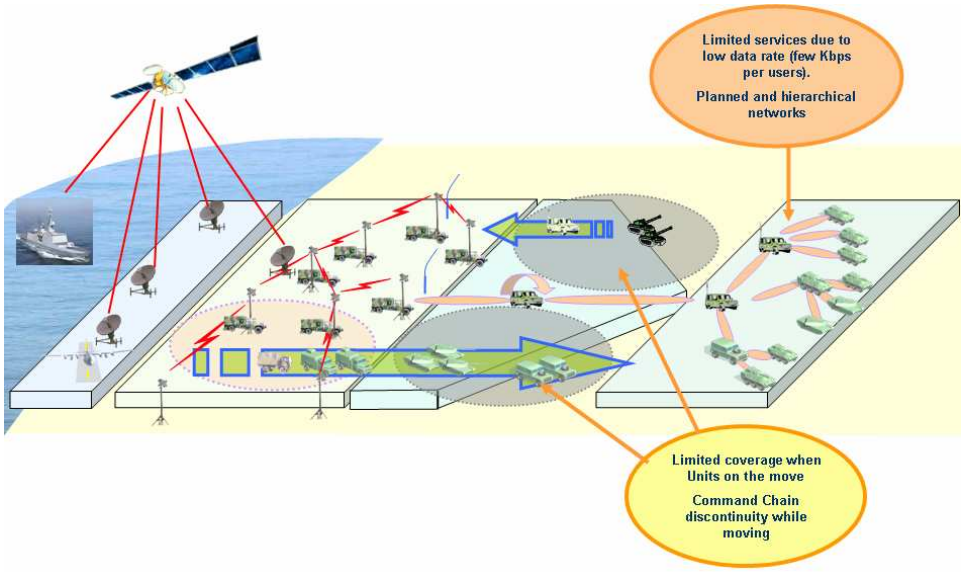

Figure 1.1 Today communications at the tactical edge

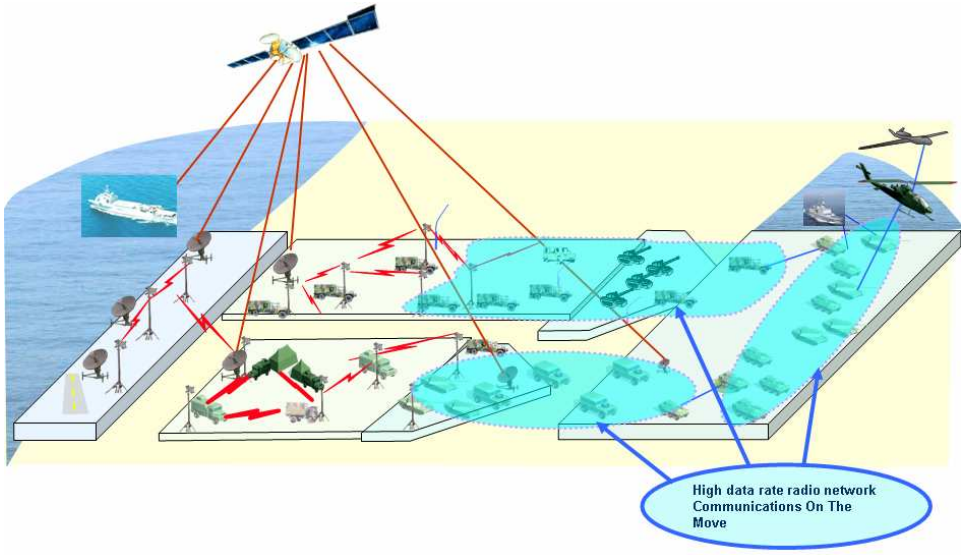

Figure 1.2 Future communications at the tactical edge

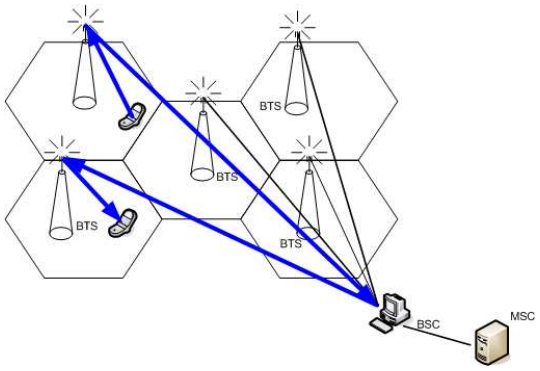

Figure 1.3 Illustration of a cellular wireless network

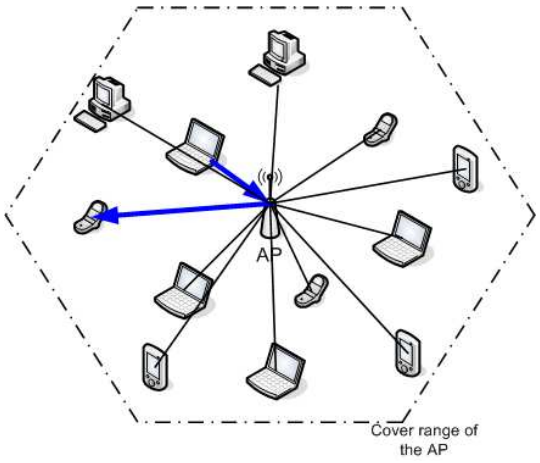

Figure 1.4 Illustration of a WIFI wireless network

funded. At the hands of such a research activity on this emerging area, the Internet Engineering Task Force (IETF) creates the Mobile Ad hoc Network Working Group (WG) [84] with the objective to standardize IP-based functionalities like routing within both static and dynamic topologies with increased dynamics due to node motion and other factors.

Unlike networks of mobiles (fig. 1.3) that use hertzian cellular networks and rely on a fixed base transceiver station or wireless networks (fig. 1.4) that are based on one of the 802.x standards and rely on a fixed access point [3], a Mobile Ad hoc Network (fig. 1.5) is characterized by the fact that it does not rely on any infrastructure. A MANET is a collection of mobile, self-configuring, self-organizing users that communicate over bandwidth constrained wireless links without relying on any infrastructure. Each node acts as a router to allow the information to go from one node to another even if these two nodes are not directly linked. The network is decentralized meaning that all network activity including discovering the topology, delivering messages, adapting to topological changes must be executed by the nodes themselves. Indeed, since nodes are mobile, the network topology may change rapidly and unpredictably over time.

Compared to other wireless networks, Mobile Ad Hoc Networks have special

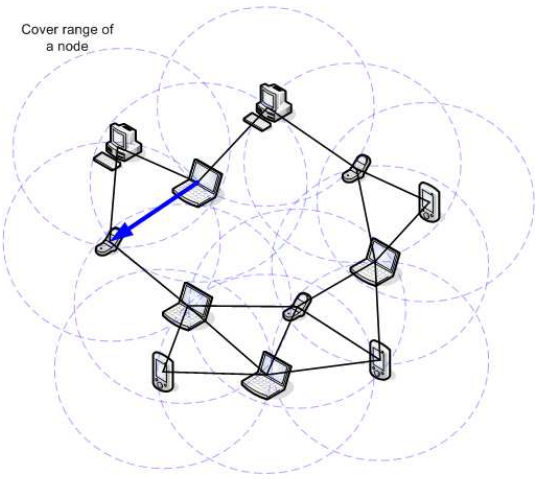

Figure 1.5 Illustration of a wireless mobile ad hoc network

features that engender some challenges [30].

- Multi-hop communications: to achieve multi-hop communications, an efficient routing service must be provided by each node. Indeed, due to the decentralized nature of the network, each node is responsible for finding the best route for the data it receives.

- Dynamic topology: the network topology is in general dynamic, because the connectivity among the nodes is time-varying due to node departures, node arrivals, and node mobility. Therefore, to be able to maintain multi-hop connectivity across the network, the routing protocol must react to topological changes.

- Bandwidth constrained and variable capacity links: wireless links have significantly less capacity than hard wired ones. Moreover the realized throughput of wireless communication is often less than ratio's maximum transmission rate due to fading, noise, multiple access, interference conditions ... This characteristic implies that congestion is more the norm than the exception. Furthermore, ad hoc networks are often the extension of a fixed network infrastructure. Thus, the customer's needs in term of traffic in the wired domain remain the same in the wireless domain becoming far more restrictive. Unfortunately, these needs will keep on increasing as multimedia and collaboration applications rise. Therefore, control applications such as QoS and routing must generate as few overhead as possible in order to offer the largest possible bandwidth to user traffic.

- Limited battery power: some of the nodes rely on battery or other exhaustible means for energy. For these nodes, an important system design criteria may be energy consumption.

- Limited security: MANETs are generally more prone to security threats such as eavesdropping, spoofing, denial of service, man-in-the-middle attacks due to the over the air medium. Nevertheless, the decentralized nature of network

control provides to MANETs robustness against the single point of failure of centralized approaches.

1.3.2 The specificities of tactical MANETs

Compared to commercial MANET, a tactical MANET presents several specificities that may turn into design objectives or challenges. For example, as far as QoS is concerned, a military user will accept intelligible voice whereas the commercial user compels a high-quality of reception [111].

1.3.2.1 Integration and interoperability

The Tactical Internet comprises a number of subsystems, among them tactical MANET, that must be integrated together efficiently. Therefore, the tactical MANET must be inter-operable with other tactical networks, with commercial networks and with networks and systems of different countries. Moreover, as opposed to traditional commercial networks where a MANET node is a host generating the user's data traffic, in tactical MANET the user or the host is not merged with the MANET node which is a radio communication equipment. Indeed, communication services are provided to users through one of the user's terminals of the TCN which is interfaced to the tactical MANET node. The user's terminals may take different forms. It may be a single terminal incorporating voice and data services, or it may be a simple data terminal such as laptop computers, notebook computers, or a multimedia terminal incorporating voice, data and possibly videoconferencing, or finally a LAN with multiple of the previous mentioned devices and possibly several routers. This architecture differs significantly with the definition of a MANET provided in [30] where a MANET is defined as a stub network to a larger fixed network infrastructure where host and routers are typically the same device. The definition of a tactical MANET is a transit network where nodes act as routers providing mostly forwarding services. As a transit network, a tactical MANET will have to interconnect with other networks running different protocols for routing, QoS, security ... services.

1.3.2.2 Scalability

Tactical MANET will exist in maritime, airborne and ground domains. The maritime domain will consist of maritime vessels, tactical edge aircraft landing from maritime vessels and amphibious vehicles. The airborne domain will consist of military aircrafts including wide-body aircrafts or Unmanned Aerial Vehicles (UAVs). Finally, the ground domain will consist of portable but stationary operation centers, ground vehicles and pedestrian soldiers. The ground domain will be the largest in term of number of nodes. Indeed, tactical MANET will typically be employed at the level of battalion and below leading to a network size that may reach thousands of routers [46]. These large network deployments differ from the commercial MANET deployments that expect no more than a few hundreds of nodes.

A tactical MANET must be able to operate in a wide range of operational deployments. The spectrum of operation may vary from a high density mechanized

operation to a low-density peacekeeping operation. Moreover, the mobility may vary from the low mobility of a walking man to the high mobility of an UAV.

1.3.2.3 Unicast

The majority of the communications in a tactical MANET are point-to-point communications. Therefore, a MANET node must implement a unicast routing protocol. The research in the field of unicast routing in MANET is very bountiful. An IETF WG [84] has been installed with the aim to standardize the IP routing protocol functionality suitable for wireless routing application in both static and dynamic networks. Five Request For Comments (RFC) have been approved by this WG of which four deal with unicast routing protocols. Therefore, it seems that the unicast routing functionality for MANET is a subject that has widely been studied. That is why, we will mainly focus on the multicast routing service as explained in the following part. Nevertheless, we will study the scalability of unicast routing protocols later in the thesis since it is a subject that still has to be covered in the field of unicast routing in MANET.

1.3.2.4 Multicast

Many military applications such as group voice, situational awareness, collaboration, video-conferencing or network management are group applications requiring the support of a multicast capability within the tactical network. Multicasting is the transmission of packets to a group of one or multiple hosts identified by a single destination address. An IP multicast service provides a way to achieve efficient communications among a group of users since it drastically limits the bandwidth consumption compared to the use of unicast or broadcast services. A much higher proportion of data in the military than in commercial world will be multicast in nature. Consequently, the multicast service requires a special attention in the tactical MANET.

1.4 Objectives of the thesis

As described in the previous section, tactical networks have several specificities that lead to several unusual challenges. The purpose of this thesis will be to define a solution to handle multicast communications in the tactical network environment. Even if we focus here on the routing issues, we should not forget that the multicast service is integrated in a network environment where multiple services are provided. The network architecture that will have to be taken into account is illustrated by the fig 1.6. The tactical network is made of local LANs connected together or to External IP Networks through the tactical MANET that acts as a transit network. Multicast communications can occur between end users belonging to the tactical network or to External IP Networks. The multicast communications will transit through the tactical MANET. Defining a multicast solution for the tactical network environment must take into account the key architectural drivers described hereafter.

- Bandwidth: The bandwidth is a scarce resource in tactical MANET. Therefore, a design goal must be for each control service deployed in the MANET to reduce

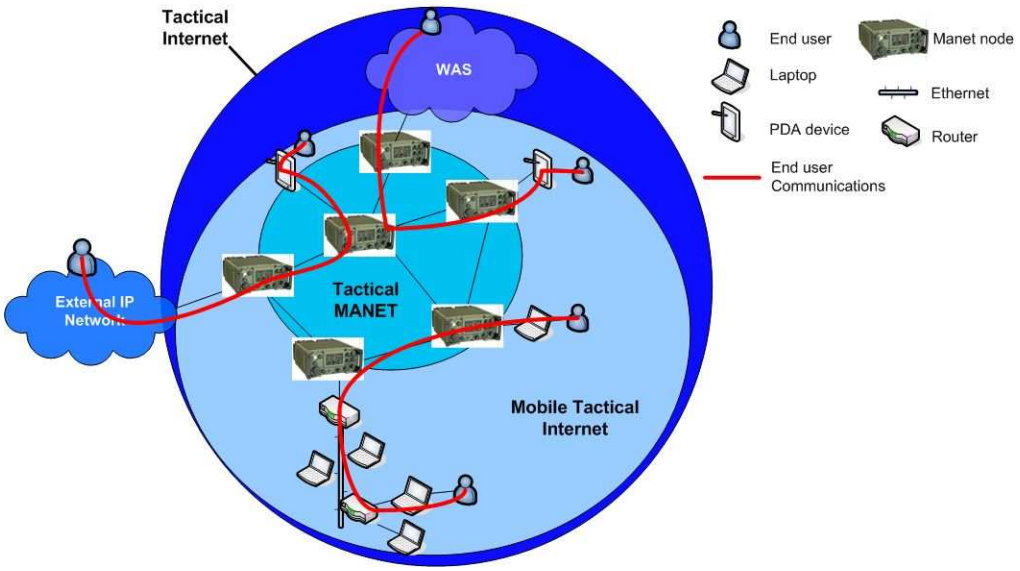

Figure 1.6 Network architecture of the tactical network environment

the control overhead needed for its operation so that the useful bandwidth for the data is as large as possible.

- Mobility: The mobility may range from the pedestrian speed i.e. low mobility to the UAV speed i.e. high mobility.
- Scalability: Large tactical MANET will contain thousands of nodes. Since a node can be connected to one or several end users, it means that the number of participants in the multicast communications can be as important as the number of nodes in the network. The multicast service must therefore be scalable with the number of nodes in the network, with the number of multicast groups and with the number of multicast participants.
- Interoperability: The solution proposed will have to provide seamless connectivity among the External IP Networks, the tactical MANET and the local LANs connected to the MANET nodes.

1.5 Organization of the thesis

This thesis focuses on defining the communication architecture and the protocols needed to provide multicast services to the tactical environment through the tactical MANET. The background and motivations have been defined previously in this chapter. The remainder of this dissertation is organized as followed.

Chapter 2: In this chapter, the protocol architecture of the multicast service within the tactical network is studied. Three solutions to achieve seamless multicast communications are considered. We show that the proxying solution is the better approach to provide an efficient multicast service in the tactical network environment. This choice underlines the need for a dedicated and specific multicast routing

protocol within the tactical MANET. Since scalability is a major constraint in this environment, strategies to provide scalability in MANET are studied and compared. The clustering approach which gathers nodes into groups is chosen. Using a clustering approach in the MANET to provide scalability underlines the need for two levels of multicast routing protocols. This will be discussed in chapter 3 and 4.

Chapter 3: The first level refers to the multicast communications within each cluster. A review of the existing multicast routing protocol for flat ad hoc networks is performed. We propose a new classification more suited to our objectives to class the existing multicast routing protocols. This taxonomy takes the design objective (robustness, efficiency, energy-saving) as the key criterion. Based on this review, we underlines a lack of protocols that can meet our requirements. Therefore, we propose a protocol called Shared-Tree Ad hoc Multicast Protocol (STAMP). This protocol defines a multicast routing algorithm based on a shared tree that is both robust and efficient. STAMP benefits from the broadcast capability of the medium to deliver data on the shared-tree similarly to a mesh-based protocol. The results of the performance evaluation of STAMP compared to the well-known ODMRP protocol are presented.

Chapter 4: The second level refers to the multicast communications between the clusters. This issue is studied in this fourth part. Through the review of the state of the art of the protocols designed to handle multicast communications between clusters (inter-cluster multicast routing protocol), we underline that all these protocols re-apply the flat techniques at the cluster level. We argue that such a solution generates to much unecessary control overhead. Therefore, we propose a new protocol, Scalable Structure-Free Inter-cluster multicast Routing (SAFIR), that does not rely on this technique. SAFIR is integrated with the other protocols deployed within the MANET and thus benefits from the other services control messages to send the information needed for its operation. SAFIR is designed to operate in association with an intra-cluster routing protocol such as STAMP. When studying the performance of SAFIR, we pay particular attention to the association SAFIR/STAMP which we compare to the SAFIR/ODMRP association. We also define how SAFIR can be integrated with any intra-cluster multicast protocol and we focus on the integration with STAMP.

Chapter 5: In this chapter, the issues related to the interoperability between the multicast routing protocols defined for the tactical MANET, i.e. STAMP and SAFIR and the multicast protocols that may be deployed in external IP networks and local LANs are studied. Different challenges are identified and solutions to these challenges are proposed. We propose mechanisms adapted for STAMP and SAFIR so that a MANET node can gather membership information from the local LAN connected to it; so that a gateway can gather membership information from the tactical MANET; so that a MANET node receives all the multicast traffic initiated by a source belonging to its local LAN; so that a gateway receives the multicast traffic initiated by any source in the tactical network. We also propose solutions for the gateway advertisement and the multiple gateway handling issue.

Chapter 6: In this thesis, we analyze the multicast service in a tactical network. This study induces us to analyze the multicast routing protocols in a clustered MANET. The multicast routing protocols we propose are based on an underlying unicast routing protocol which brings to the fore the need for a scalable unicast

routing protocol in the tactical MANET. The review of the litterature of protocols designed to enhance and adapt the well-known Optimized Link State Routing (OLSR) unicast routing protocol to a clustered network highlights that all these protocols propose to leverage the mecanisms of OLSR at the cluster level. We argue that such a solution generates to much control overhead. Therefore, we propose a solution to adapt OLSR to a clustered mobile network where a regular version of OLSR is applied within each cluster for the intra-cluster communications but where inter-cluster communications are realized thanks to the definition of a new message sent by the clusterheads. Contrary to the other existing protocols, our solution does not apply a version of OLSR on the cluster topology. Theroretical and simulation analyses prove that our protocol outperforms its counterparts.

Chapter 7: This last chapter gives a summary of the achieved work and concludes the thesis. A discussion on the future directions of work in the topics addressed by the thesis is performed.

Chapter 2

Architecture of the Multicast Communication Service

This chapter presents the architecture of the multicast communication service that must be set up to handle seamless end-to-end multicast communications between hosts belonging to External IP Networks and LANs or Ethernet segments of the TCNs forming the mobile tactical network through the tactical MANET. Firstly, the IP multicasting service defined for wired IP networks is presented. It defines the architecture and the protocols that may be deployed in the IP networks interfacing with the tactical MANET. Then, the multicast-oriented tactical network structure is defined. In the third part, three possible approaches to interconnect the multicast service of different IP networks through the tactical MANET are presented. Finally, the last part presents the structure of the multicast service in the tactical MANET which is driven by a major constraint, the scalability.

2.1 IP multicasting in wired IP networks

Traditionally, in the Internet, data exchanges concern two hosts which communicate through unicasting, for example using the client/server paradigm. Multicasting is a special type of communications where one host identified as the source wants to send the same data to a group of hosts identified as the members. Groups of receivers are said to participate in multicast sessions. Applications for such a type of communications are numerous, video on demand, video-conference, database update ... Although multicasting has been a desired technology for some time now, protocols are still evolving and standardization is still in progress.

Indeed, to achieve the distribution of data to several receivers, three opportunities may be possible as described below.

- broadcast: The first solution (figure 2.1) to address a group of hosts in the network is to send the data to all the hosts even those that do not want to receive them. It is then each host responsibility to choose to accept or to discard the data. A single copy of the data is sent by the source and upon reception; each network component duplicates it and forwards a copy to all its neighbors. This solution is not optimal since the data is duplicated many times in the network, wasting an important bandwidth. Moreover, it imposes additional processing to the receiver and it may disturb needlessly many network equipments and hosts. Broadcast is thus a solution that consumes uselessly network resources as well as the resources of recipients that do not want to receive the data.

- unicast: The second solution (figure 2.2) is to send a copy of the same data to each destination host in turn. A unicast communication is thus established between each pair source/destination. This solution imposes that the source knows the list of the destinations. Moreover, it wastes bandwidth since as many copies of the same data as destination hosts are sent over the network. This solution is inefficient since it consumes both sender and network resources in term of processing power and memory.

- multicast: This third solution (figure 2.3) constructs a delivery structure to connect all senders and receivers. A single copy of the data is sent over the network and this copy is duplicated in the network at determined components on the delivery structure, when paths diverge.

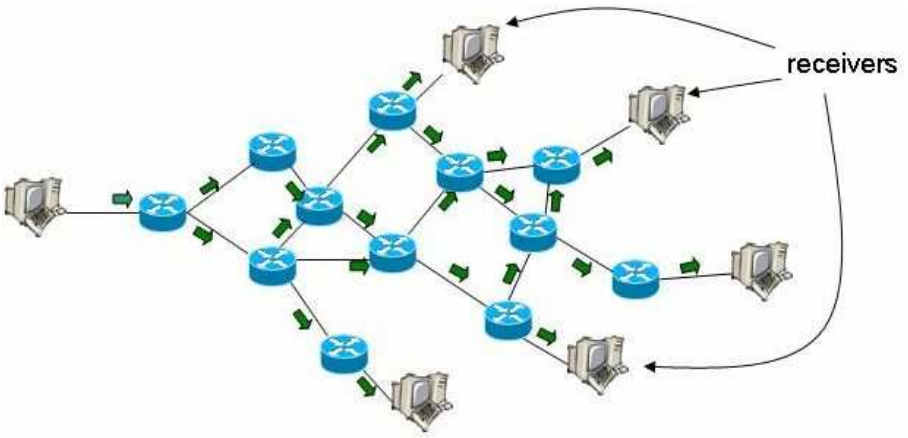

Figure 2.1 Multicast communication via a broadcast service

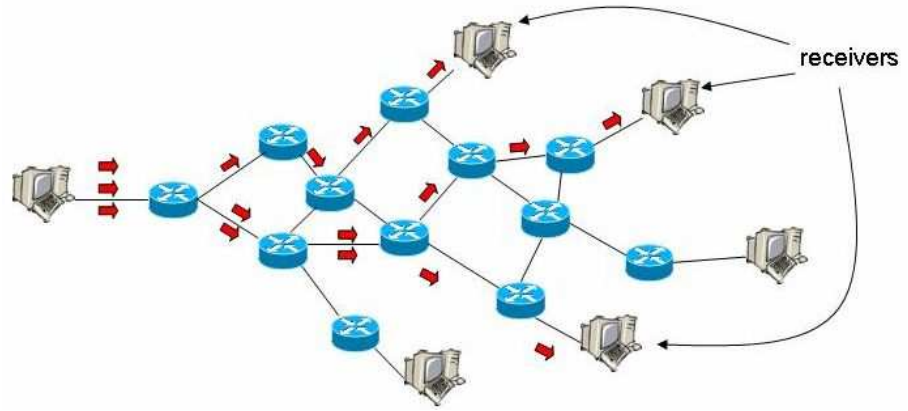

Figure 2.2 Multicast communication via a unicast service

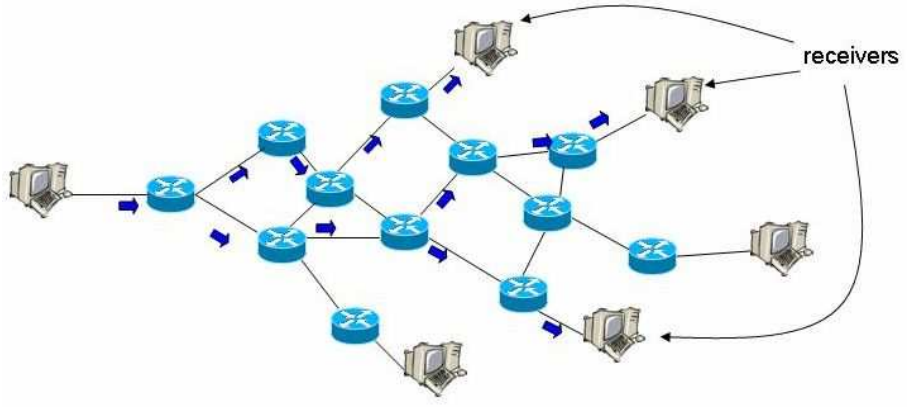

Figure 2.3 Multicast communication via a multicast service

The Internet multicast model The concept of multicast was first introduced by Steven Deering during his PhD [35, 36]. This concept has not known an immediate keen interest. A reason for this is that multicasting requires additional "intelligence"

within the network which is at the odds with the Internet belief that wants to push this intelligence to the edge of the network (see the predominance of Diffserv on Intserv for example). In his dissertation, Deering defined the multicast model. Multicast refers to the transmission of an IP datagram to a "host group". A host group (or multicast group) is a set of hosts that want to receive the same information. The participants of this group are also called the members. The "multicast membership" of a set of hosts refers to the list of multicast groups for which at least one of the hosts is a member. IP datagrams are delivered to all members of the multicast group with the same best effort reliability as unicast IP datagrams.

The membership of a host group is dynamic, i.e. the hosts may leave or join the group whenever they want. There is no restriction regarding the location of the hosts or the number of hosts in a group. A host may be member of several groups at once. A host does not need to be a member of a group to send data to it, i.e. the source of traffic for a group does not need to belong to the group. There may have several sources for a group. A multicast group is assigned a unique IP multicast address (IP Class D address) ranging from 224.0.0.0 to 239.255.255.255.

The network model (figure 2.4) to send multicast traffic is composed of hosts (the actual members of the group) and of their local router that connects them to the Internet. This network model is enriched by two protocols, a membership management protocol and a multicast routing protocol. The local routers and the hosts employ a group membership management protocol, for example the Internet Group Management Protocol (IGMP) [34] to exchange group membership management information about the multicast group membership. Each receiver registers to its local router for one multicast group by sending it its will to join the multicast group. Registering for a multicast group means that the local host wants to receive the data addressed to this group. When a host does not want to participate in the multicast session anymore, it also informs its local router. IGMP provides thus the final hop in the multicast delivery service since it only manages the multicast traffic distribution between the local router and the group of hosts directly attached to it.

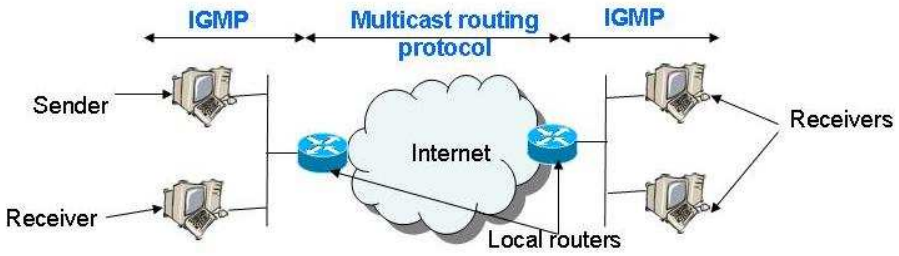

Figure 2.4 Network Model Introduced For Multicast

A multicast routing protocol is run in the Internet by the routers in order to enable an end-to-end multicast service. The multicast routing protocol is responsible for the construction and the maintenance of the multicast delivery structure and also for the efficient forwarding of the multicast packets. When a multicast router receives an indication from one of its host saying that the host wants to be a member of a multicast session, it means for the router that it must belong to the delivery structure maintained for this multicast group. Therefore, as far as the mul-

ticast routing protocol is concerned, the router becomes a member for the multicast group. One can notice that there is a distinction between the notion of member for the multicast membership management protocols and the notion of member for the multicast routing protocols. For the former, the members are hosts and these hosts are the actual destinations, the sinks of the multicast traffic. For the latter, the members are routers and these routers are the local routers of the hosts that are actual members. The routers are not the sinks for the multicast data.

In order to transmit the multicast traffic through the network, the routers construct thanks to the multicast routing protocol a delivery structure that is known as a Spanning Tree that connects all the members (in the multicast routing protocol meaning) of a group. This tree contains the minimum number of routers so that there is only one path between every pair of routers and no routing loop. Thus, if a router knows which of its interfaces belong to the spanning tree, it can copy an incoming multicast packet to all the interfaces belonging to the spanning tree, except the incoming one. That way, only the minimum needed number of copies is generated and the messages are replicated only on the tree branches which minimizes the number of copies of the messages that are transmitted through the network. The spanning tree is updated depending on the group membership dynamics. Branches on which there is no member anymore must be pruned or discarded. In the spanning tree, there is an upstream/downstream or parent/child relation between routers. As illustrated by the figure 2.5, the node that is the next hop on the path back to the initiator of the spanning tree is the upstream or parent node and a node that can be reached through an outgoing interface is a child or downstream node. In the example, B is the parent or upstream node of C and C is a child or downstream node of B. Note that a node has only ONE upstream node or parent but can have one or several children or downstream nodes.

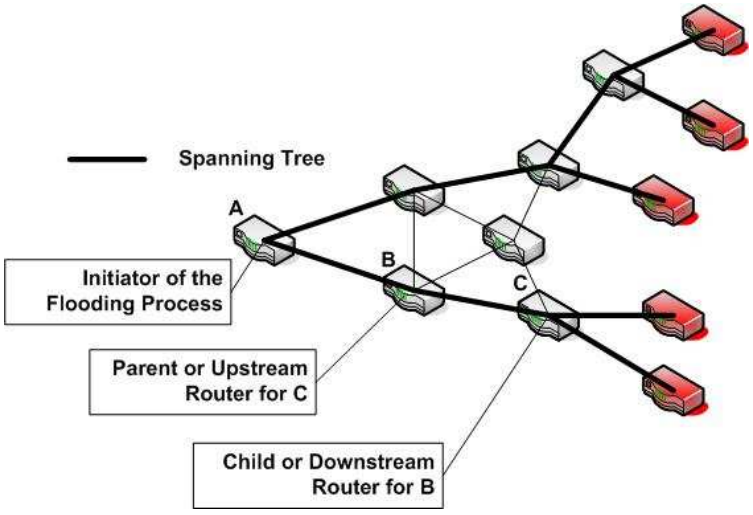

Figure 2.5 Example of a spanning tree and illustration of the parent/child or upstream/downstream relation

The way the spanning tree is constructed and how the routers interact depends on the objectives of multicast routing protocol. Thus, the multicast routing protocols follow two approaches called Dense Mode (DM) and Sparse Mode (SM).

- Dense Mode: this approach assumes that the members of the multicast groups are densely distributed among the network and that the bandwidth is plentiful. This means that almost all routers in the network will need to distribute traffic for the multicast groups. Therefore, the most efficient way of constructing the spanning tree is to include everyone in it through an initial flooding of the network and then to cut or prune the useless branches (branches without receiver) which may be few.
- Sparse Mode: the sparse mode approach assumes that few routers in the network are involved in the multicast groups and that the bandwidth is scarce as it is the case in the Internet for example. Therefore, in the Sparse Mode approach the distribution tree is initially empty, and the needed branches are added one by one as the result of an explicit joining process that uses in general the unicast routing table information. No flooding is used since it would represent an important waste of bandwidth in such a network environment.

Due to the division of the global Internet in domains, two categories of protocols have been specified, the intra-domain multicast routing protocols and the inter-domain multicast routing protocols. Intra-domain multicast routing protocols are employed by multicast-enable routers to allow multicast communications within a domain of the Internet, i.e. an Autonomous System (AS) [56]. The principal intra-domain multicast routing protocols proposed by the IETF and by the research community are the Distance Vector Multicast Routing Protocol (DVMRP) [127], the Multicast Open Shortest Path First (MOSPF) protocol [92], the Core Based Tree (CBT) protocol [10], the Protocol Independent Multicast (PIM) - Dense Mode (DM) [1] or - Sparse Mode (SM) [40]. The inter-domain multicast routing protocols are used to allow multicast communications across the different domains and are implemented by the border routers. Examples of this type of protocols are the Border Gateway Multicast Protocol (BGMP) associated with the Multicast Address-Set Claim (MASC) protocol [70], the Multicast Source Discovery Protocol (MSDP) [41], the Explicit Requested Single Source (EXPRESS) multicast protocol [57] and the Simple Multicast (SM) protocol [11]. Among all these protocols, only some of them are implemented or deployed, i.e. PIM-SM, PIM-DM, DVMRP and MOSPF for the intra-domain multicast routing protocols and MSDP and BGMP/MASC for the inter-domain multicast routing protocols. The others are "research" propositions.

Description of the protocols The annexe 1 provides brief descriptions of some of the protocols involved in the Internet Multicast model, beginning by the IGMP protocol. Then, it presents three intra-domain and two inter-domain multicast routing protocols.

2.2 Multicast-oriented tactical network structure

The architecture of the tactical network has been presented in the first chapter of the thesis and illustrated by figure 1.6. In this architecture, the tactical MANET is interconnected with several networks, the LAN of each TCN, the WAS and the other External IP Networks (commercial IP networks, tactical networks of other countries...). All these networks can be classified in the three following categories with respect to the type of structure they carry out for the multicast service.

- Ethernet Segment: It is composed exclusively of hosts. The ad hoc node is directly connected to the hosts. The IGMP protocol is employed by the hosts to report the multicast membership to their local router which is the MANET node. This Ethernet Segment refers to the Ethernet segment that may compose a TCN in case it does not contain any router.
- LAN: It is composed of hosts and routers. The hosts employ the IGMP protocol with their local router which may or may not be the MANET node. The routers of the LAN implement a multicast routing protocol to construct a multicast delivery structure. The MANET node is linked to these routers. These LANs refers to the networks of each TCN or to the WAS.
- External IP Network: It can be any type of IP networks such as the global Internet or a military network. The MANET node is connected to routers which implement an inter-domain and/or an intra-domain multicast routing protocol.

One can argue that the LANs and the External IP Networks are redundant. Nevertheless, we need to distinguish these two types of networks since LANs can be considered as "proprietary" networks whereas External IP Network are networks on which no control is possible. External IP Networks can be considered as made of COTS equipments whereas LANs are made of equipments on which assumptions, controls or requirements can be imposed.

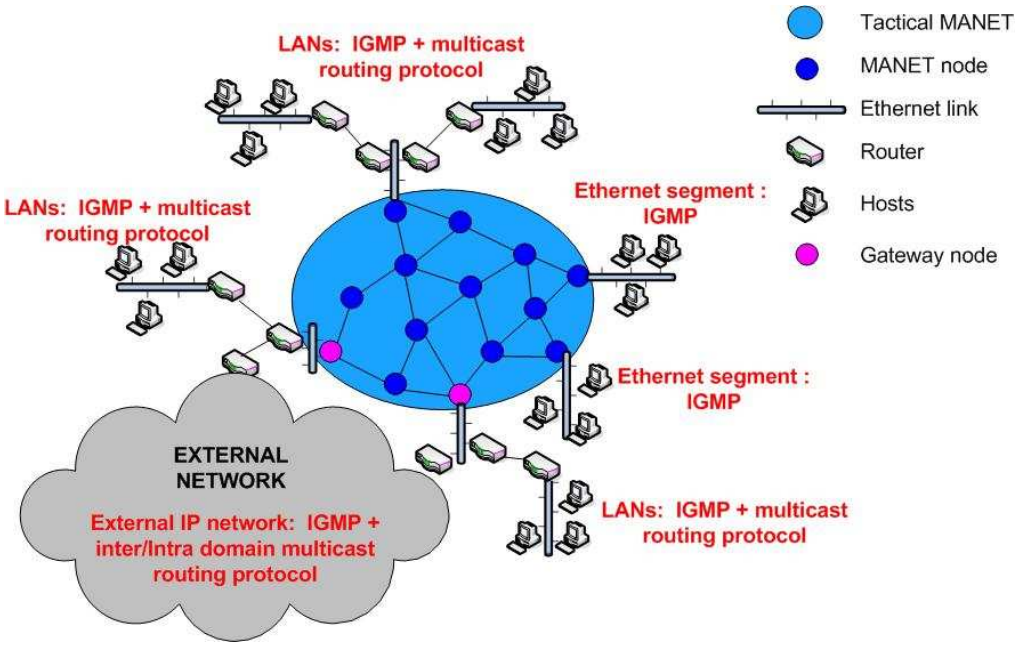

Figure 2.6 Multicast-oriented network architecture

This classification leads to the concept of *gateway node*. A gateway node is a MANET node that provides interconnection with an External IP Network. Figure 2.6 illustrates the preceding classification and the concept of gateway node.

The multicast members can be located in any of the three types of networks. Therefore, the MANET may be used to interconnect different configurations of net-

Table 2.1 Network interconnections considered for multicast service

	Ethernet Segment	LAN	External IP Network
Ethernet Segment	√	√	√
LAN	√	√	√
External IP Network	√	√	×

works employing different multicast protocols, an External IP network with a LAN, a LAN with a LAN, an Ethernet Segment with a LAN... Nevertheless, we will not consider interconnection of two External IP Networks through the MANET network. Indeed, we consider that External IP Networks should have other possibility of interconnection than the MANET network. All other possibilities will be considered as illustrated by the table 2.1.

2.3 Multicast interconnection of wired IP networks and MANET: three solutions

In the context of the tactical networks, the tactical MANET is a transit network, carrying traffic which enters and then leaves the network. This traffic is generated by hosts belonging to external IP networks, LANs or Ethernet Segment attached to the tactical MANET nodes. This network architecture is different from the one considered in commercial networks where the MANET is seen as a stub network, meaning that all traffic must be either sourced or sinked by a MANET node. The multicast members, either sources or members, may belong to any type of networks.

Considering the multicast service, the MANET nodes are not responsible for the decision to be part of any multicast group, but rather the hosts that belongs to the Ethernet segment, the LANs or to the External IP Network. The tactical MANET can be seen as a backbone of routers. A MANET node have a dual IP stack (at least) with a wired interface on the Ethernet segment, the LANs and the External IP network if it is wired IP network and a wireless interface on the tactical MANET network.

If the architecture for the multicast service in IP networks is well defined[1], a solution still has to be proposed to interconnect these architectures through the tactical MANET to provide seamless end-to-end multicast connectivity to multicast participants. Indeed, the types and the number of networks that a multicast connection goes through must be transparent to the end multicast users.

Three solutions can be considered to provide seamless multicast service through the MANET network:

- If the MANET nodes are non multicast-aware routers: the tunneling
- If the MANET nodes are multicast-aware router implementing wired multicast protocols: End-To-End seamless IP Multicasting

[1]IGMP for the Ethernet Segments, both IGMP and a multicast routing protocol for the LANs and intra/inter domain multicast routing protocols for the External IP networks

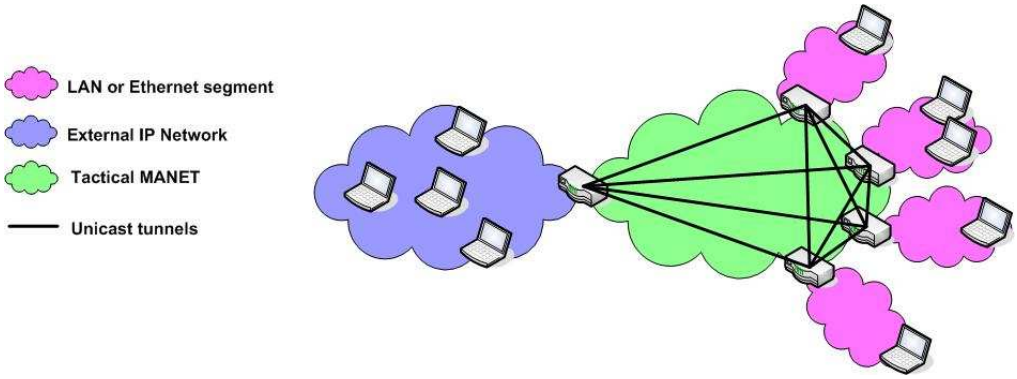

Figure 2.7 Illustration of the tunneling solution

- If the MANET nodes are multicast-aware routers implementing specific MANET multicast routing protocols: the proxying

Each of these solutions are described and analyzed hereafter.

2.3.1 The tunneling

The architecture that can be set is directly linked with the multicast capability of the MANET nodes. If the MANET nodes are non multicast-enable nodes on their MANET interface, the only solution that can be envisaged is to create multicast tunnels over the tactical MANET following the example of what is done in the MBone with the non multicast-enable routers. The tactical MANET and its attached networks belong to a different multicast domain than the external IP Network. The External IP Networks are considered to belong to different multicast domains. All LANs and Ethernet Segments associated with their local router can either be seen as multiple multicast domains or as a single multicast domain. In the former case, the MANET is an independent multicast domain where routers do not have any multicast capabilities. In the latter, the MANET nodes form a backbone of non multicast-capable routers in a multicast domain. In both case, unicast tunnels are created between the LANs or the Ethernet Segments themselves, and also between the LANs or Ethernet Segments and the External IP Networks as illustrated by the figure 2.7. For example, if the External IP Network employs the BGMP protocol, tunnels are created between the border routers implementing this protocol. The tunnels are set up through the MANET network.

This approach is not scalable and presents performance shortcomings. Indeed, potentially, each LAN or each Ethernet Segment connected to a MANET node may present multicast members for a particular multicast group. Therefore, the number of unicast tunnels may grow with the size of the MANET network which may reach several hundreds of nodes. This arises a problem of control overhead since the number of unicast tunnels and thus of control connections may grow with a factor N^2 where N is the number of nodes in the MANET. Moreover, since each data packet is unicasted into each tunnel, the data overhead may also be very penalizing especially for a MANET where the bandwidth is a scarce resource.

2.3.2 End-to-End seamless IP multicasting

The tunneling solution presented in the previous section is far from optimal and may generate a significant overhead in such a bandwidth-constraint environment. Therefore, even if the MANET is only a transit network for the multicast communications, the MANET node must be multicast-enable in order to avoid the tunnels. Considering that the MANET nodes are multicast-enable, the first idea that comes in mind is to apply the well established multicast routing protocols defined for the wired Internet. The MANET can be defined as an independent multicast domain. MANET nodes must therefore implement an intra-domain multicast protocol such as PIM-SM, MOSPF or DVMRP and an inter-domain multicast routing protocol such as BGMP or MSDP. If the MANET, the Ethernet Segments and the LANs form a unique multicast domain, then the multicast protocol implemented in the MANET should be consistent with the one implemented in the LANs. In this case, the MANET gateway nodes may also be required to implement an inter-domain multicast routing protocol.

In the context of MANET, the multicast routing protocol needs to manage not only the group member dynamics (nodes joining or leaving a group) but also the dynamics of the node location since nodes are mobile. However, the multicast routing protocols employed in the wired Internet has been designed for fixed networks. For example, with the protocols employed and designed for the wired Internet, a node can not re-emit a multicast data packet on the interface through which the packet has been received. This concept is not applicable in the MANET context, since a MANET node generally only has one wireless interface through which it receives and then re-emit the packet if needed. Moreover, they may fail coping with the topology changes and the node movements that may be faced in a MANET. Adapting to node movement and topological change means re-building the multicast structure any time a change occurs which may result in a substantial control overhead. As a conclusion, multicast routing protocols designed for the wired Internet are not adapted to the MANET environment. Consequently, it appears that the best solution for the multicast service in a MANET consists in implementing a multicast routing protocol specifically designed for MANET. It is the proxying solution.

2.3.3 The proxying

It has been shown that the MANET nodes must be multicast-capable to provide an efficient multicast service and that they must implement a multicast routing protocol that is designed specifically for the MANET environment. This protocol must propose an efficient maintenance process to cope with the evolution of the topology. It must also propose processes that do not generate too much control overhead since the bandwidth is scarce. Finally, the amount of information that each node must store should be reduced due to the limited storage capacity of MANET nodes.

Since we propose that a specific multicast routing protocol is employed within the MANET as illustrated by figure 2.8, translating and proxying mechanisms should also be defined to interconnect with the traditional wired IP multicast protocol that will be employed in the LANs, in the Ethernet Segments and in the External IP Networks. Each local LAN or Ethernet Segment will be seen as an independent

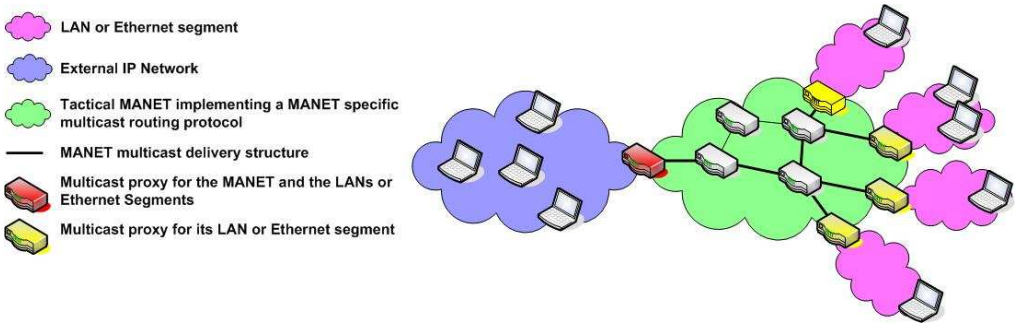

Figure 2.8 Illustration of the proxying solution

multicast segment for its local ad hoc node and therefore, will not have the vision of the other LANs or Ethernet Segments. Moreover, with respect to the External IP Network, the MANET, the LANs and the Ethernet Segments will be seen as a unique multicast domain. Therefore, the local ad hoc nodes will act as multicast proxies for their local LANs or Ethernet segments, and the gateway ad hoc nodes will act as multicast proxies for the whole tactical Internet.

Proxying raises some interoperability issues that will be discussed in the chapter 5 of this document. This chapter will discuss how the wired IP multicast solutions will interact with the MANET multicast routing protocol to provide seamless end-to-end multicast connectivity. It will also discuss how the multicast memberships will be exchanged between gateways and/or local proxies. Nevertheless, before considering such issues, the multicast routing protocols employed in the MANET must be defined.

2.4 Structure of the MANET multicast service and the scalability driver

We showed that in order to provide an efficient end-to-end seamless multicast service through the tactical MANET, the multicast service within the tactical MANET should be provided through the implementation of a MANET-specific multicast routing protocol. Designing a multicast routing protocol in a MANET environment is a complex problem due to the fact that the group membership can change and also that the network topology may highly evolve causing link failures. Moreover, the limited bandwidth availability coupled with the limited energy resources make the design a challenge.

2.4.1 Requirements for the multicast routing protocol in a tactical MANET

The basic solution to perform multicast distribution in a MANET environment is through flooding. Nevertheless, this approach results in poor efficiency in term of bandwidth utilization. An efficient approach should design a structure that covers only the group members. The major requirements when designing a multicast routing

protocol are the following [91, 93]:

- Robustness: The mobility of nodes in MANETs causes frequent link failures which result in packet dropped and therefore in a poor delivery ratio. A multicast routing protocol is said to be robust if it is able to achieve high packet delivery ratio even under a high node mobility;
- Efficiency: One of the main characteristic of Mobile Ad hoc Networks is the scarce bandwidth. Hence, the efficiency, defined as the ratio of the total number of data and control packets transmitted in the network to the total number of data packets received by receivers, is of great importance. The efficiency measures the amount of control and data overhead sent through the network. A multicast routing protocol is said to be efficient if it does not generate too much control and data overhead;
- Control overhead: To construct and maintain the multicast structure, to keep track of the multicast members, the multicast routing protocol exchanges control messages. These processes should be optimized so that the control message overhead does not occupy the whole bandwidth and thus does not prevent data transmission;
- Quality of service: For applications such as Situational Awareness, voice, video, QoS is very important. Thus provisioning QoS must be taken into account when designing a multicast routing protocol;
- Resource management: MANETs are made of limited battery power and memory nodes. The operations that are energy consuming, i.e. the transmission of packets, the reception of packets, the storage of information and the computation of information should be reduced;
- Security: As for QoS, the military/strategic applications are applications that need a high degree of security. The multicast routing protocols should be able to provide security mechanisms such as authentication of session members, prevention of non-members from gaining unauthorized information...
- Scalability: The tactical environment brings to the fore the need for a scalable multicast routing protocol. Indeed, the number of nodes in a tactical MANET can range from a squad size (around 10 nodes) to the brigade size (several thousands of nodes) [13]. This characteristic shall be put into perspective with the commercial applications of MANETs such as conference, classroom, where the size of the network is not expected to rise above one or two hundreds of nodes. Therefore, multicast routing protocol employed in the tactical MANET should be able to scale to networks with a large number of nodes. Another aspect of the scalability requirement deals with the number of multicast groups and the number of participants (both source and receivers) in each multicast group.

Designing a multicast protocol is thus a great challenge. Nevertheless, it is obvious that a multicast protocol can hardly satisfy all the above requirements. The most restrictive and challenging requirements which is also the one that has the most important impact on the multicast service structure is the scalability with the network

size. Indeed, the way scalability with the network size can be achieved in a MANET can influence the structure of the network and thus is a driver for all the services provided in a MANET and particularly the multicast service as we will see in the next section.

2.4.2 Scalability in MANET: the current propositions

Scalability can be broadly defined as whether the network is able to provide an acceptable level of service to packets even in the presence of a large number of nodes in the network [102]. The first researches on scalability for MANET have been done in the field of unicast routing [59, 114]. A network composed of a high number of nodes represents a challenge for the unicast routing protocols since the more nodes, the more routes to compute and store, and the more control traffic to send ... All the more that when the number of nodes increases, the available bandwidth for control data decreases. The first protocols proposed for routing in MANET have been designed to be optimal in "small" networks. Therefore, their control overhead, their storage capacity and their processing capacity grow proportionally with the number of nodes in the network. Simulation works have shown that these traditional routing protocols can hardly support network with more than 300 nodes. Therefore, propositions to handle scalability have been made.

Due to the fact that the bandwidth is scarce in MANET, the scalability issue for a protocol in MANET concerned mostly the overhead of control messages which increases with the network population and the mobility. Moreover, the storage capacity is also critical since for example with the unicast routing, large routing tables imply a large control packet size and hence a large link overhead. Thus we will consider that a scalable routing protocol is a protocol that aims at reducing the control overhead (O/H), the storage capacity and the processing capacity.

Several solutions have been considered to provide scalability. All these approaches have in common the will to reduce the control O/H. To achieve this goal, they use either the network density, the communication capacity of nodes ... We are going to distinguish the protocols thanks to the nodes uniformity. Uniform protocols treat the nodes uniformly i.e. consider that there is no hierarchy in the network, all nodes send and reply to routing control messages in the same way. Uniform protocols providing scalability features are the reactive protocols. Non uniform protocols make distinction between the different nodes in order to reduce the control traffic burden. Non uniform protocols fall into two categories: protocols in which each node selects a subset of the network and focus control activities on it and protocols that divide topologically the network. The first approach is called "neighbor selection approach" and the second one "hierarchical approach".

2.4.2.1 Reactive approach

The reactive approach follows the idea that control message overhead could be reduced if each node only sends its control messages when needed, for example when a communication is awaiting for unicast routing. This approach is different from the proactive one, where periodic control messages are exchanged so that each node always has the information about all the services, even if it does not need it, for example, the routes to all nodes in the network for a unicast routing protocol. Nodes

running proactive protocols only have to store the information they need. Reactive protocols are characterized by a discovering phase during which control packets are flooded over the network. This phase ends when the needed service is discovered, for example the address of the gateway node. If the service to be discovered changes during the discovery process, the discovering phase needs to be re-initiated.

Even if the on-demand characteristic of this approach limits the control overhead since there is no periodic message, reactive protocols have drawbacks that prevent them from being scalable to large networks. The control overhead of the discovering phase is directly linked to the number of nodes in the network and also to the number of nodes that need the service. For the unicast routing, the control overhead is proportional to the number of communicating pairs. In networks with more than thousand nodes, communications pairs are expected to be numerous, generating an important overhead. Moreover, if the service is lost because of node mobility (if the route breaks for example), the discovering process need to be re-initiated. In case of highly mobile networks, link breakages may append often. In term of latency, since the service has to be discovered before using it, reactive protocols introduce an initial latency that is growing as the network is going larger or the diameter increases. Experiments performed on reactive routing protocols [8], Dynamic Source Routing (DSR) [66] and Ad hoc On-demand Distance Vector (AODV) [100], show that the delivery rate falls to 20% for networks of more than 300 nodes when there is mobility because the control overhead occupies most of the bandwidth. As a conclusion, reactive approaches do not scale to networks of more than few hundreds of nodes, and so are not good approaches for tactical MANET.

2.4.2.2 Neighbor selection approach

Unlike reactive protocols, the neighbor selection protocols are non uniform, which means that a distinction between the network nodes is made when receiving or sending control messages. Neighbor selection approaches select a set of nodes in the network to which the control messages are forwarded or which are the forwarders of a received control message. Those nodes are selected either on the distance between the source and them with the Fisheye approach [99] for example, or upon their position and the connectivity in the neighborhood of the sending node with the Multi-Point Relay (MPR) approach [61] for example.

The Fisheye solution is inspired by the "fisheye" technique proposed by Kleinrock and Stevens [68], where the technique was used to reduce the size the information require to represent graphical data. The fisheye technique consists of frequently forwarding information to nearby nodes while reducing the frequency as the destination is farther. Thus, as the eyes of a fish capture with high details the pixels near the focal point, the nodes near the sending one have more accurate information than farther nodes.

The goal of Multi-Point Relays is to limit the flooding of broadcast packets in the network by minimizing the duplicate retransmissions locally. In the multi-point relay approach, a node selects a subset of neighbors called the Multi-Point Relays (MPRs) to retransmit broadcast packets. This allows neighbor nodes which are not in the MPR set to read the message without retransmitting it.

In spite of the scalability efforts made, neighbor selection protocols still have characteristics that prevent them from being fully scalable with the network size.

As far as the storage and processing capacity are concerned, each node still has to store and compute information about all other nodes in the networks. In case of a unicast routing protocol, each node has to store and compute a route to all other nodes in the network. In case of mobility leading to topological changes and links breakages, control information need to be exchanged to update the local information creating additional overhead. Moreover, in case of a topological change in an area of the network, even if it is a local change, it is the entire network that has to update its information. Focusing on the MPR concept, it has proved to be efficient only in high density networks. Indeed, when the network becomes sparse, all neighbor nodes are selected as MPR and the optimization is null.

2.4.2.3 Hierarchical approach

The problem of scalability has also been encountered in the wired Internet. The solution proposed to cope with scalability was to introduce hierarchy by dividing the network into Autonomous Systems [56]. Indeed, having levels of hierarchy allows each node to have only a local view of the network and thus reduces the storage capacity, the processing capacity and also the control overhead. For example, the growth of the size of the routing table is only logarithmic rather than linear with the number of nodes.

Considering this statement, several hierarchical protocols have been proposed for use in the MANET environment. These solutions can be distinguished from all the preceding ones which are said to be "flat". In flat solutions, no distinction between nodes is made. The wireless hierarchical protocols are based on the idea of organizing nodes in groups and then assigning nodes different functionalities inside and outside a group. Nodes are gathered into groups, called clusters, where a particular node in the cluster called the clusterhead represents the entire cluster for nodes outside of the cluster. The clustering process allows each node to store all the information relating to its cluster and only a part of the information regarding other clusters. That way, the control overhead is limited since both the number of messages and the size of the messages are decreased, the storage and processing capacity are reduced and the impact of the mobility is only local.

The advantages of the clustering for the routing purpose are obvious, but this technique is also interesting for other services. For example, it can ease the resource sharing and/or the synchronization within a cluster, it can also ease the network management and it may allow the spatial reuse of the radio frequencies to minimize interferences [80]. It has also been proved that clustering improves the stability [101] and the capacity [53] of the network. The main drawback of the clustering technique is that the maintenance of such structures needs a robust mechanism that may generate overhead. Nevertheless, the clustering solution is the most promising technique in response to the scalability challenge faced by tactical MANET. Moreover, in the field of tactical networks where nodes often move as groups, the clustering technique allows to match the physical architecture with the logical network architecture defined by the clustering protocol.

Many clustering protocols for MANET have been proposed in the literature. It is not the point of this dissertation to provide a state-of-this-art of clustering in MANET. To get more information on this topic, the reader can refer to [137]. Among all the proposed protocols for clustering in MANET, a first category is made by the 1-

hop clustering protocols [19,22,38,49,80]. These protocols divide the network so that each node is at maximum 1-hop away from a clusterhead. Such algorithm produces a large number of clusters when the size of the network increases. Moreover, each topological change at the node scale implies a reconstruction of the whole structure. The second category of protocols is the k-hops clustering protocols [5,44,69,81,103]. With these protocols, each node is at maximum k hops away from its clusterhead. K-hops clustering solutions lead to a number of clusters that is inferior to the 1-hop solution presenting therefore better scalability features. Moreover, the whole structure does not need to be re-built at each topological change.

In conclusion, it appears that the clustering approach and more precisely the k-hop clustering solution is the most promising solution to handle scalability in tactical MANET. Indeed, the hierarchy introduces in the network through the clustering technique has a positive impact on the control overhead of the routing protocol since nodes of the network only need a local view of the network. Moreover, the size of the routing table is reduced as well as the influence of a link breakage which only has a local impact. Finally, clustering also presents benefits for other services such as the network management. Consequently, we will then considered that the tactical MANET is divided into clusters through a clustering protocol resulting in a network structure such as the one illustrated by figure 2.9.

Concerning multicast communications in the MANET, this network structure made of clusters has repercussions on the multicast service. Indeed, when the clustering process builds the clusters, the multicast group members repartition is not taken into account. Therefore, several situations may happen:

- all sources and members of a multicast group belong to the same cluster.
- all members of a multicast group belong to the same cluster and the sources belong to different clusters.
- Members and sources are spread over several clusters.

Depending on the repartition of the actors of the multicast communications, the multicast service may be limited to a single cluster or may span over several clusters. Therefore, we have to distinguish two levels of multicast communications:

- The intra-cluster communications which refer to how the nodes that belong to the same cluster can exchange multicast information among themselves.
- The inter-cluster communications which refer to how the multicast data can be exchanged between the different clusters (that is to say how a node that belongs to a specific cluster can exchange multicast traffic with other nodes that belong to other clusters).

To handle these two levels of communications, a possible solution would have been to design a global protocol. Nevertheless, in order to make benefits from the clustering structure and also because the two levels of communications have different constraints, we choose to consider and thus optimize them separately. Consequently, we will distinguish two multicast routing protocols, an intra-cluster multicast routing protocol that will handle intra-cluster multicast communications and an inter-cluster

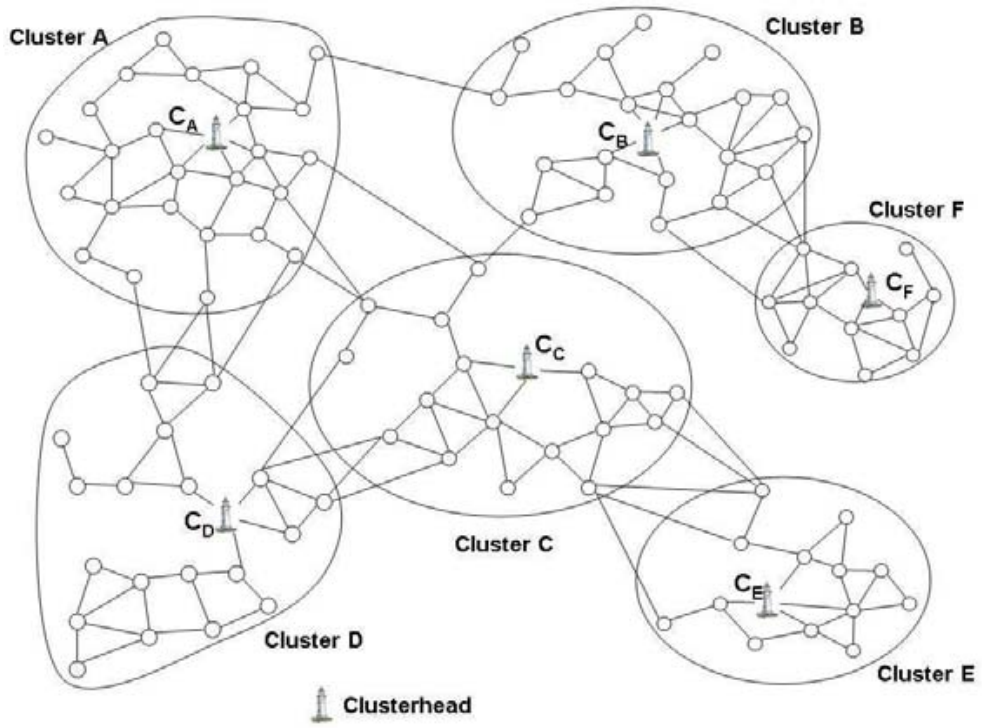

Figure 2.9 Illustration of the clustered tactical MANET

multicast routing protocol that will handle the inter-cluster multicast communications. Chapter 3 presents our solution regarding the intra-cluster multicast routing protocol and chapter 4 focuses on the inter-cluster multicast routing protocol.

Chapter 3

Intra-Cluster Multicast Routing Protocol

Tactical MANET are special types of wireless networks where the number of nodes may reach several thousands of nodes. As presented in chapter 2, clustering is the most promising solution to handle the well-known scalability issues encountered by routing protocols in such large scale networks (size of routing table, control O/H generation ...). The network is then divided into clusters with a leader in each, called the "clusterhead". Multicast communications in such hierarchical clustered networks lead to two distinct challenges. The first challenge is to achieve multicast communications inside each cluster and the second challenge is to achieve multicast communications between those clusters. In this chapter, we will focus on the first challenge referred as the intra-cluster multicast communications.

The size of a cluster depends on the chosen clustering protocol and on the network topology. Nevertheless, we can expect that a cluster may be made of a number of nodes that is compatible with the use of the multicast routing protocols designed for commercial purposes that rely on a flat network topology. Therefore, the majority of the multicast routing protocols proposed over the past ten years that focus mainly on multicast communications into flat networks, can be candidates for the intra cluster operations.

In a first part, we are going to review the requirements for the intra-cluster multicast routing protocols. Then, we will present a review of the multicast routing protocols proposed in the literature. Based on the analysis of the state-of-the-art of flat multicast routing protocols, we propose the Shared-Tree Ad hoc Multicast Protocol (STAMP). In the last part, we will present the results of the performance evaluation of STAMP done with the OPNET discrete event simulator.

3.1 Requirements on the intra-cluster multicast protocol

The principal requirements that are expected on the intra-cluster multicast routing protocol are described hereafter.

- Robustness: The mobility of nodes in MANETs causes frequent link failures which result in packet dropped and therefore in a poor delivery ratio. A multicast routing protocol is said to be robust if it is able to achieve high packet delivery ratio even under a high node mobility;

- Efficiency: One of the main characteristic of Mobile Ad hoc Networks is the scarce bandwidth. This constraint is even more important in tactical MANET where the bandwidth is often less wide than in commercial MANET. Hence, the efficiency, defined as the ratio of the total number of data packets received by receivers to the total number of data and control packets transmitted in the network, is of great importance. The efficiency measures the amount of control and data overhead sent through the network. A multicast routing protocol is said to be efficient if it does not generate to much control and data overhead;

- Control overhead: To construct and maintain the multicast structure, to keep track of the multicast members, the multicast routing protocol exchanges control messages. These processes must be optimized so that the control message

overhead does not occupy the whole bandwidth and thus does not prevent data transmission;

- Energy Consumption: MANETs are made of limited battery power and memory nodes. The operations that are energy consuming, i.e. the transmission of packets, the reception of packets, the storage of information and the computation of information should be reduced;
- Quality of service: For applications such as Situational Awareness, voice, video, QoS is very important. Thus provisioning QoS shall be taken into account when designing a multicast routing protocol;
- Reliability: Some of the information sent by military users are sensitive information that need a high degree of reliability. Mechanism to provide reliability features may be provided to the multicast routing protocol.
- Security: As for QoS, the military/strategic applications are applications that need a high degree of security. The multicast routing protocols should be able to provide security mechanisms such as authentication of session members, prevention of non-members from gaining unauthorized information...

Note that the scalability requirement is no more applicable to the intra-cluster multicast routing protocol compared to the requirements described in section 2.

Designing a protocol that satisfies all these requirements may be made in several steps. Moreover, some of these requirements do not rely exclusively on the multicast routing protocol but rather on a cross-layering approach where the multicast routing protocol is only a piece. The QoS requirement is one of these special requirements. Therefore, in this thesis, we choose to focus on the robustness, the efficiency, the control overhead and the energy consumption. In future works, we will focus on integrating the multicast routing protocol in a QoS scheme and on adding the security functionalities to the robust, efficient and energy saving protocol we choose at this first step.

3.2 Review of the flat MANET multicast routing protocols

Research in the field of multicast routing for Mobile Ad hoc Network has been particularly prolific over the last decade. It is roughly one hundred protocols that have been proposed. Consequently, some taxonomy or methods of classification have also been envisaged in order to compare and class all these protocols. First, we review some of the existing taxonomies. Then, based on the requirements described previously, we analyze these taxonomies and the proposed multicast routing protocols.

3.2.1 Several taxonomies to class multicast routing protocols

Basically, the objective of a multicast protocol is to construct and maintain a sort of structure that will enable a multicast data packet to be delivered from the sources to

the destinations. In the traditional wired Internet, the classification of the proposed protocols is trivial since there are only two classes of multicast routing protocols either based on a tree routed at the source or on a shared tree. In the context of multicast for MANET, the classification is less obvious. Indeed, even if the first propositions of multicast protocols for MANET were based on the traditional wired tree-based approaches, a wide diversity of propositions has arisen among the last ten years in order to cope with the different constraints of the MANET environments. Therefore, multiple taxonomies have been envisaged including the topology of the structure, the route acquisition scheme, the initialization of the multicast session, the dependency on unicast routing, the topology maintenance mechanism and the structure connectivity.

3.2.1.1 Topology of the structure

In addition to the classical tree based approaches coming from the traditional wired multicast routing protocols, a second type of multicast topology has appeared for multicasting in MANET: the mesh-based topology. In contrast to tree-based approaches where only a single path exists between a source/receiver pair, mesh-based protocols (On Demand Multicast Routing Protocol -ODMRP- [77, 136], Core Assisted Multicast Protocol -CAMP- [47], Forwarding Group Multicast Protocol [25], Dynamic Core-based Multicast routing Protocol -DCMP- [32], Neighbor Supporting Multicast Protocol -NSMP- [76]...) may have multiple paths between any source and receiver pair. Tree-based multicast protocols have proved to be more efficient compared to mesh-based since tree-based protocols transmit the minimum number of copies of the information whereas mesh-based send duplicate copies through redundant paths. Nevertheless, mesh-based multicast protocols are more robust due to the availability of multiple paths between the source and the receiver which allow multicast datagrams to be delivered to the receivers even if links fail.

Tree-based multicast protocols can be further divided into two types: source-tree-based (Multicast Optimized Link State Routing -MOLSR- [72], Associativity-Based Ad hoc Multicast -ABAM- [122], Adaptive Demand-driven Multicast Routing -ADMR- [63] ...) and shared-tree-based (Multicast Ad hoc On-demand Distance Vector -MAODV- [108], Ad hoc Multicast Routing utilizing IncreaSing ID numbers -AMRIS- [130], Lightweight Adaptive Multicast -LAM- [64], RObust Multicasting in Ad hoc Networks using Trees -ROMANT- [124]...). In source-tree-based multicast protocols, the tree is rooted at the source whereas in shared-tree-based protocols a single tree is shared by all sources within the multicast group and is rooted at a node referred as the core node. The source-tree-based multicast protocols perform better under heavy loads because of an efficient traffic distribution among links while the shared-tree-based multicast protocols are more scalable. The main problem with the shared-tree-based approach is that it heavily depends on the core node, and hence, a single point of failure at the core affects the performance of the multicast protocol.

3.2.1.2 Route acquisition scheme

This taxonomy comes directly from the unicast routing field. Proactive approach protocols compute paths to all multicast destinations in advance whether or not the paths will be used to forward data. The information is stored in the nodes and

is available as soon as it is needed by any application. Reactive approach protocols attempt to provide an on-demand service to application. The operation of the protocol, i.e. the construction of the multicast paths is driven by the needs of the applications, by the presence of a multicast data packet to transmit. These two antagonistic approaches had opposed for many years. Now, a sort of consensus has been found, saying that each solution has network conditions to which it is more suited.

The advantage of a proactive approach is that there is no latency at the initialization of a multicast data transfer since the multicast delivery structure is already available whereas it has to be firstly constructed before sending data with the reactive approaches. The advantage of the reactive approaches is that they engender less control overhead since structure are only constructed and maintained when there is data to send. There is no waste of bandwidth by creating and maintaining a structure that will never be used. Nevertheless, network-wide flooding is often employed in reactive approach firstly to construct and then maintain the structure as long as the multicast session is running. Moreover, with the reactive approach, the amount of control overhead is directly linked to the network and applications characteristics. One should note that reactive protocols often employ proactive mechanisms to maintain the structure for example through periodical testing of neighborhood or through periodical reconstruction of the structure.

3.2.1.3 Initialization of the multicast session

The multicast group formation and the construction of the multicast delivery structure (tree or mesh) can be initiated by the source as well as by the receivers. If the initiation is made by the source node, the protocol is called a source-initiated multicast routing protocol and if it is up to the receivers to initiate the group and the construction, the protocol is called a receiver-initiated multicast routing protocol.

The source-initiated approach could also be called traffic-demand multicast routing protocol. Indeed, in this approach, member nodes keep silent when they become group receivers. Each source is responsible for announcing itself to the network. A source generally announces itself for the first time when it has data to transmit. Then, after this first announcement, the source either periodically announces itself even if it does not have data to transmit any more or does it on-demand. A reactive protocol is often a source-initiated protocol since it is the source protocol that has the information of the multicast session start time and that can initiate the construction of the structure.

The receiver-initiated approach could also be called group-demand multicast routing protocol. In this approach, the group members have the responsibility for initiating the construction of the multicast delivery structure. Receiver initiated protocols are often based on a shared structure where the core node is one of the members. With a sender-initiated protocol, control messages may be sent in the network even if there is no multicast data packet being sent. Indeed, when a node becomes a member and initiates the construction of a multicast structure, it cannot know if there is any source in the network, unless the protocol imposes to source nodes to announce themselves to the other nodes in the network just to let them know that they exist. This behavior is not optimal since there is no need for a source to announce itself if the protocol is not sender-initiated; it is a waste of bandwidth.

In some protocols, both sources and receivers have no clear distinction and are treated equally as group members. Generally these protocols exploit similar mechanisms than those used by receiver-initiated protocols.

3.2.1.4 Dependency on unicast routing

Contrary to the case of Internet, in MANET, the researches in unicast and multicast routing have nearly begun at the same time. This is the reason why we can find protocols that rely on a unicast routing protocol and protocols that do not rely on any existing protocol. According to this dependency, we can classify the protocols into three categories:

- Protocols that are totally dependent: the multicast routing protocol discovers group receivers and the unicast protocol provides routes to the concerned receivers and is also responsible for the forwarding of the data packets (their performance depends on unicast protocol one's) e.g. DDM (Differential Destination Multicast [65]);

- Protocols that are partially dependent: these protocols need information from the unicast protocol (typically Next Hop information) to construct and maintain delivery structure but the packet forwarding is done by the multicast routing protocol via the delivery structure (requires less unicast information than the first solution). Examples of this type of protocol are CAMP (Core-Assisted Mesh Protocol [47]), LAM (Lightweight Adaptive Multicast [64]);

- Protocols that are totally independent: the multicast protocol itself realizes all functionalities multicast and unicast (needs to periodically probe the network and thus reacts slowly to topology changes). Often, these protocols are adaptation of existing unicast protocols that have been completed to provide the multicast capability. Examples of this type of protocol are MAODV (Multicast Ad-hoc On demand Distance Vector [108]), AMRIS (Ad hoc Multicast Routing protocol Utilizing Increasing Id numbers [130]), ODMRP (On Demand Multicast Routing Protocol [77]), and MOLSR (Multicast Optimized Link State Protocol [72]).

3.2.1.5 Topology maintenance mechanism

Once the delivery structure (tree or mesh) is constructed, the multicast routing protocol must maintain it. This maintenance may be done either by a soft state approach or by a hard state approach. In the soft state approach, control information is exchanged periodically to prevent link breaks, membership modification i.e. to update the tree or the mesh. The states stored in the nodes are associated with timers. In the hard state approach, control information is only transmitted when a link breaks. This second solution results in lower overhead. In the paper defining the ST-WIN protocol which is one of the first adaptation of the PIM-SM protocol for wireless networks [23], the authors compare the control overhead of the soft-state and hard-state version of their protocol. They show that even if the control overhead of the hard-state version increases with the mobility; it remains lower than the soft state version. With the hard state approach, the information of link

breakages may come from the MAC layer or from the unicast routing protocol that may employ a Hello message mechanism for example to keep updated information about its neighborhood.

3.2.1.6 Structure connectivity

The delivery structure may be dedicated to a single source or may be shared by all the sources of a multicast group. In the first case, the delivery structure is said to be a source-based structure. Such structures are made of the unions of all the shortest-paths from the source to all receivers. It is thus made of optimal paths and proposes minimum end-to-ends delays. Moreover, since the nodes belonging to the structure may differ from one source to the other, source-based approaches can ease load balancing. However, since each source has to construct and maintain its own structure, it can yield important control overheads and memory resource consumptions. Such approaches are well suited for networks where the number of sources is small and where the applications are time-critical. On the other side, group-shared protocols create and maintain only one structure per group whatever the number of sources is. These protocols are well suited to networks where the number of sources is important and where each member can also be a source. Nevertheless, these approaches present the limitation to concentrate the traffic on some nodes which may cause congestion in case of high traffic load.

3.2.1.7 Conclusion

Based on all these taxonomies, a classification of some of the protocols proposed in the literature is presented in table A.1.

Multicasting in a MANET is a multi-variable issue. The topology must be robust but the number of radio resource employed must be kept minimal, the physical medium used must be optimized to limit superfluous transmission and retransmission, the amount of control message must be minimal. Moreover, the storage capacity as well as the complexity of the algorithm must be low to preserve node battery. Since it is difficult to optimize in the same algorithm several constraints, most of the proposed protocols focus on optimizing one of the requirements. Our objective is to find a protocol that is robust, efficient and energy saving.

Among all the preceding taxonomies, the one based on the topology structure presents the greatest interest due to the fact that it is the one that has the most global view of the protocol. Nevertheless, this taxonomy does not take into account the emerging approaches for multicasting in MANET such as the overlay, the location aware, the energy efficiency ... Those approaches are different in the sense that they exploit a characteristic of the MANET network or that they focus on one of the design objectives of a multicast routing protocol, the energy saving for example.

Consequently, it comes that none of the proposed taxonomies allows to determine if one protocol responds to the design objectives we have. We propose to have a more global and practical view of the existing protocols and therefore to classify them based on the design objective that is favored.

Concerning the protocols proposed in the litterature, it would have also been interesting to pay attention to the way the forwarding of the packet is performed, and how the multicast routing protocol operate with the MAC layer and the unicast

Table 3.1 Comparative Table of the Multicast Routing protocols

	Topology	Route Acquisition	Initialization	Unicast Dependency	Topology Maintenance	Structure Connectivity
ABAM [122]	Source Tree	Reactive	Traffic	Independent	Hard State	Source
ADMR [63]	Source Tree	Reactive	Traffic	Independent	Soft State	Source
AMRIS [130]	Shared Tree	Reactive	Elected Source	Independent	Soft State	Group
AMRoute [12]	Shared Tree	Proactive	Both	Partially Dependent	Soft State	Group
ASTM [24]	Shared/Source Tree	Proactive	Both	Partially Dependent	Soft State	Source and Group
BEMR [96]	Mesh	Proactive	Both	Independent	Hard State	Group
CAMP [47]	Mesh	Proactive	Both	Partially Dependent	Soft State	Group
CQMP [37]	Mesh	Reactive	Source	Independent	Soft State	Group
DCMP [32]	Mesh	Reactive	Source	Independent	Soft State	Group
DDM [65]	Source Tree	Reactive	Source	Dependent	None	Source
DPUMA, PUMA [123]	Mesh	Proactive	Group	Independent	Soft State	Group
FGMP-SA [25]	Mesh	Proactive	Source	Partially Dependent	Soft State	Source
FGMP-RA [25]	Mesh	Proactive	Group	Partially Dependent	Soft State	Group
GBMP [134]	Shared Tree	Reactive	Source	Independent	Soft State	Group
LAM [64]	Shared Tree	Proactive	Both	Partially Dependent	Hard State	Group
MANSI [117]	Mesh	Reactive	Source	Independent	Soft State	Group
MAODV [108]	Shared Tree	Proactive	Both	Independent	Soft State	Group
MOLSR [72]	Source Tree	Reactive	Source	Independent	Soft State	Source
MRDC [131]	Shared Tree	Reactive	First Source	Independent	Soft/Hard State	Group
MSTP [98]	Shared Tree	Hybrid	Source	Independent	Hybrid	Group
MZRP [133]	Source Tree	Hybrid	Source	Independent	Soft State	Source
NSMP [76]	Mesh	Reactive	Source	Independent	Soft State	Group
ODMRP [77]	Mesh	Reactive	Source	Independent	Soft State	Group
ODMRP-MPR [138]	Mesh	Reactive	Source	Independent	Soft State	Group
ODMRP-PDA [14]	Mesh	Reactive	Source	Independent	Soft State	Group
ROMANT [124]	Shared Tree	Proactive	Group	Independent	Soft State	Group
SMMRP [55]	Mesh	Reactive	Source	Independent	Soft/Hard State	Group
SRMP [90]	Mesh	Reactive	Source	Independent	Soft State	Group

routing protocol to forward the packets. For example, we could have studied if the protocol employs unicast transmission, or encapsulation, or if the protocol needs a duplicate packet detection function. Nevertheless, in the environment where STAMP will be deployed, the multicast will be treated as a broadcast by the MAC layer and a duplicate packet detection method is applied to each packet sent over the network. Therefore, such caracteristics would not have been discriminatory for the choice of a multicast routing algorithm.

In the following section, we will review the multicast routing protocols for MANET by classifying them depending on the design objective they focus on. We will also integrate in this review the recent emerging approaches.

3.2.2 Design objective-based state-of-the-art

The different criteria we will consider have been presented in the paragraph 3.1.

3.2.2.1 Robustness

The robustness criterion measures the capacity of the protocol to go on operating even in case of mobility, of topological changes. The robustness requirement is a requirement that appears for multicast routing protocols in MANET, which is directly linked to the variable topology characteristic of this type of networks. Indeed, in the wired Internet, a topological change caused by a link breakage is a rare event. Mesh-based approaches have been specifically designed to solve this issue. Indeed, the first approach for multicast is the tree structure which is very sensitive to topological changes. When a link breaks, data packets may be lost until the branch is repaired since there is only one path between each source/receiver pair. The proposition of the mesh-based approach is to introduce redundancy by providing several paths from sources to receivers.

The most known mesh based protocol is ODMRP [77]. The mesh-based protocols present one major drawback: the data overhead is high since several copies of a data packet are sent among different paths. This point may be critical for applications with a high throughput. Moreover, the initialization of the protocol may also be an issue for mesh-based protocols. Indeed, most of the mesh-based protocols use a source-initiated approach and thus flooding to send the first data or the control packets and also to periodically refresh the mesh (soft state approach). Therefore, these protocols present an important control overhead that increases with the number of sources. Several simulation studies [74, 78] confirm this statement. Some propositions have been made to solve this issue. For example, CAMP [47] or PUMA (Protocol for Unified Multicasting through Announcements [123]) have taken a receiver-initiated approach, using core to initiate the structure. That way the control overhead due to the initialization of the structure is avoid. Other protocols, such as NSMP [76], ODMRP-MPR (ODMRP-Multi Point Relay [138]), ODMRP-PDA (ODMRP-Passive Data Acknowledgement [14]) propose modifications of the ODMRP protocol to reduce the control overhead caused by the maintenance of the mesh. For instance, NSMP employs a local route recovery mechanism to reduce the overhead of route failure recovery and mesh maintenance. NSMP also attempts to reduce data transmissions.

Another solution that is studied to design robust multicast protocols is the location-aided protocols. This type of protocol relies on the availability of a Global Positioning System (GPS) system in each or most of the network nodes. With the GPS, each node is provided with its location and mobility information. With the GPS support, ODMRP can be enhanced to adapt to node movement. Each node can estimate the route expiration time and thus can anticipate the re-construction of routes before their breaks, making ODMRP more resilient to mobility. In general, such protocols assume that the sender node knows the list of the multicast member nodes and also their geographical position. Therefore, when forwarding a multicast data packet, the source chooses the best next hop (for example, the one that is the closest to the destination members) for each multicast member and indicate to which multicast members the packet is for. Therefore, any intermediate node can process the data packet since it knows the destination nodes from the received data packet and their geographical position. This solution presents the major advantage that it is resilient to mobility and that no structure is needed. Nevertheless, the dissemination to all network nodes of each node position as well as the identity of the multicast members may consume an important bandwidth.

3.2.2.2 Efficiency and control overhead

The efficiency of a multicast routing protocol can be measured by the amount of control packets and data packets that must be transmitted to achieve good delivery ratios. The less control packet sent and data packet retransmitted the better efficiency.

As long as the data overhead is concerned, the tree-based structure presents the best efficiency in term of data duplication since a tree is the optimal structure to link a source to multiple receivers.

Concerning the control overhead, protocols employing source initialization rely on flooding to set up and refresh the structure. Most of the mesh protocols as well as the source-tree protocols enter in this category. Moreover, protocols employing soft-state for the topology maintenance are characterized by an important control overhead since they exchange periodical messages to maintain the structure. Shared-tree protocol presents a good efficiency in term of control overhead when the mobility is low. They do not rely on any flooding for protocol operating. Nevertheless, when the mobility increases, more control messages must be exchanged to repair the tree branches.

Some new trends appear these last years in the area of multicast routing protocols for MANET. These trends focus for most of them in reducing the control overhead.

The first category to consider is the stateless multicast. The idea behind stateless multicast protocol is that since nodes are mobile, it is better to store states in the header of the multicast data than in the network nodes. Indeed, storing states in the network nodes implies to maintain the states which is bandwidth consuming particularly in case of mobility. Therefore, stateless multicast protocols do not rely on any structure. The state added in the packet header can be the list of destination for DDM [65], or a series of location coordinates for the location-guided small group communications protocols [20]. When receiving a multicast data packet, a node processes the header of the packet and based on the information contained in the header decides whether to forward the packet (where, who) or not. These approaches

suppose that the source nodes know the list of the multicast destinations. Stateless multicast protocols minimize the control overhead (no control overhead is needed except, possibly, the messages needed by the members to announce themselves to the sources). Nevertheless, the information added in the header of each data packet can be considered as control overhead. Moreover, the additional information added in each packet increase with the size of the multicast group. Finally, this overhead also increases with the traffic load since this information is added in each data packet. Therefore, these protocols are well suited to small multicast groups in large networks. Stateless multicast protocols rely on the unicast routing protocol. Therefore, their performance with respect to the mobility are directly linked to the unicast routing protocol chosen. It is expected that the protocol chosen for the stateless approach is a proactive protocol since a reactive protocol will have to maintain routes to all multicast members. Moreover, if the unicast routing protocol does not converge rapidly enough, the unicast routing table on which the multicast forwarding decision is based may be stale and the multicast data packet may be lost.

Overlay multicast also called "end system multicast" is a solution proposed to provided multicast service in the Internet. The idea is that non-member nodes do not need to be burden with multicast routing and storing state information. The multicast protocol is thus used at an application layer rather than at the network layer. That way, the multicast protocol is independent from the physical topology. The multicast member nodes form an overlay network which links are made of Transmission Control Protocol (TCP) or User Datagram Protocol (UDP) flows supported by the underlying unicast routing protocol through unicast tunnels. On top of this overlay connectivity the multicast member nodes perform the agreed multicast routing. Ad hoc Multicast Routing (AMRoute [12]), Multicast Overlay Spanning Tree (MOST [45]), Progressively Adapted Sub-Tree in Dynamic Mesh (PAST-DM [51]), Prioritized Overlay Multicast Ad hoc (POMA [132]) are examples of such protocols. AMRoute creates bi-directional tunnels to connect multicast members into a virtual mesh. Then, a shared-tree is created for data delivery and is maintained on top of this virtual topology. The fact that overlay multicast protocols are independent from the physical topology makes them efficient in term of control overhead since the structure does not have to be updated each time the topology changes. Nevertheless, the removal of routing intelligence from the network nodes may yield some penalties in term of bandwidth usage. Indeed when several overlay connections share the same physical link, it increases the link stress as illustrated by the figure 3.1. The fact that the overlay structure is not updated in reaction to the topological changes may lead to sub-optimal paths and thus to data overhead. Therefore, the efficiency in term of data overhead is not achieved. Moreover, in term of robustness, an overlay network may face degradations of performance with high degree of mobility [89]. Finally, it seems difficult to add QoS features to this type of protocols since the structure and the links are chosen independently from the physical topology.

3.2.2.3 Energy consumption

MANET nodes are supposed to be driven by limited battery resources. Therefore, it comes that designing a protocol that is energy-conserving is essential. Many works have been done in the field of optimizing the energy consumption for a broadcast delivery. In the field of multicasting, two goals are conflicting when designing an

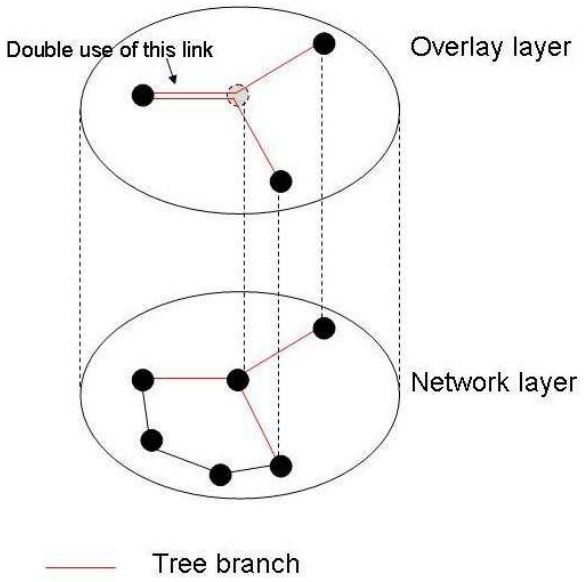

Figure 3.1 Illustration of the overlay bandwidth inefficiency

energy-efficient multicast routing protocol. On then one hand, some protocols are designed with the goal to optimize (minimize) the total energy consumption of the multicast tree. They are called the minimum energy multicast tree protocols. On the other hand, some protocols are designed to optimize (maximize) the lifetime of the multicast tree where the lifetime of a multicast tree in term of energy corresponds to the duration of the multicast service until a node fails due to its lack of energy. L-REMIT (Lifetime-Refining Energy efficient of Multicast Trees [129]) or LMT (Lifetime-aware Multicast Tree [85]) are examples of such maximum lifetime multicast tree protocols. Wireless transmission and reception are the greatest contributors to energy consumption in ad hoc networks. Then, the third contribution is given by the energy needed to perform processing. Therefore, a common idea in designing an energy-conserving protocol is to reduce the number of nodes that participate in the multicast forwarding and to reduce the amount of control packets sent. Following this idea, mesh-based protocols may perform waste than tree-based protocols in term of energy efficiency because of the initial broadcast and because of the greater size of the forwarding group that participates in the data forwarding.

3.2.2.4 QoS, reliability and security

QoS, reliability and security are issues that are encountered in multicast communications.

QoS can be defined as a set of service requirements that the network must meet while transporting a packet stream from a source to a destination. The network guarantees that it is able to satisfy a set of predetermined service performance constraints that are defined through a set of measurable attributes such as the end-to-end delay, the available bandwidth, the probability of packet loss, the delay variance ... For the MANET, we can also add the power consumption and the service coverage

area. The provision of QoS to group communications is still an open issue for wired network. The mobility of nodes in MANET adds another dimension to the problem. Therefore, QoS-aware multicast for MANET remains an open problem. Few propositions have been made so far. QAMNet [120] approach proposes to enhance the ODMRP protocol by introducing traffic priority, distributed resource probing and admission control mechanism to provide QoS multicasting. The QMR (QoS for Multicast Routing [112]) protocol proposes a flexible hybrid scheme and integrates bandwidth reservation function into a multicast routing protocol based on a mesh-topology. Recently this protocol has been enhanced [113] to integrate a cross-layer design to provide multicast QoS. In our mind, providing QoS to multicast communications needs a cross-layer approach where the MAC layer, the unicast routing, the transport protocol and other additional processes work together. Among the few propositions that can be found in the literature for Multicast QoS for MANET, the majority is based on a non-QoS protocol that is enriched thanks to additional QoS functionalities. Our approach is similar to this one since we propose to perform several spirals to define our multicast protocol. The first step of our work which corresponds to this thesis is to define a multicast routing protocol that is compliant with the robustness and the efficiency requirements. Then, in future works, we will concentrate our efforts on adding QoS features to this protocol.

Concerning security the approach is similar. Security is an essential requirement in the field of wireless networks and even more essential in the field of tactical MANET. The use of a wireless medium makes ad hoc networks prone to passive attacks such as eavesdropping and active attacks. Passive attacks allow an attacker to access secret information. Active attacks may destruct messages, inject fake messages, modify messages and impersonate a node and consequently may jeopardize the availability, the integrity, the authentication and the non-repudiation which are the basic elements of the network security. The intrinsic attributes of MANET expose such networks to additional attacks such as jamming attacks, power consumption attacks or routing attacks. Defining the appropriate architecture to secure the multicast communications is also a great challenge. This architecture may provide several services which are the data confidentiality, the forward and backward secrecy, the source authentication, the group authentication and the control access to the group members. The combination of the ad hoc network characteristics and the group communications make the security of multicast communications in an ad hoc network a great challenge. For the moment, the solution that seems to be the better suited to provide the security services needed in multicast communication architecture is to employ a Group Key Management Protocol (GKMP). This protocol must distribute a Transmission Encryption Key (TEK) to the source so that it can cipher its data and to the members to decrypt them. The GKMP is also responsible for the security of the TEK distribution through the deployment of Key Encryption Keys (KEKs) and for the renewing of the keys. The protocol must adapt to the ad hoc network characteristics. This approach does not seem to have a major impact on the multicast routing protocol itself. Therefore, the security of the multicast communications is an issue that we will consider in future works that are not part of this thesis after having defined our robust and efficient multicast routing protocol. When designing the secure multicast communication architecture, the control overhead a GKMP may generate, the time needed to cipher and decrypt the data, the

time needed for the key distribution are example of parameters that will have to be considered.

The reliability can be defined as the capacity of the protocol to ensure reliability properties with respect to the delivery of data to the destinations, as opposed to an unreliable protocol that does not guarantee that a packet will be delivered intact or that it will be delivered at all. A way to achieve reliability is to implement some error recovery mechanism. In the Internet area, some sender-initiated reliable protocols designed for small area implement the Automated Repeat Request (ARQ) mechanism. MTP (Multicast Transfer Protocol [7]) is an example of such a protocol where the sender is responsible for processing positive or negative acknowledgments and for retransmitting packets. However, since such a solution is not scalable, some other reliable multicast protocols have been proposed. These protocols allow either dedicated receivers or routers to handle ACKs/NACKs and to retransmit packets for members in their local environments. Designing reliable multicast protocols for MANET is a very challenging task. However, some protocols have been studied. On the one hand, we can distinguish the deterministic protocols (Reliable Broadcast [97], Reliable Adaptive Lightweight Multicast Protocol [119], Family ACK Tree [79]). They provide "all-or-nothing" delivery guarantee for the delivery of packets to a multicast group. On the other hand, probabilistic protocols (Anonymous Gossip [18], Route Driven Gossip [82]) provide a guaranteed delivery with a certain probability. A survey of reliable broadcast protocols for MANET [126] concludes that deterministic protocols have bad trade offs between reliability and scalability/mobility while probabilistic protocols do not provide deterministic delivery guarantees. Our point of view is that a reliable approach for multicast requires a retransmission scheme that can be added to an existing multicast routing protocol or can be implement by an additional process and also an error detection and advertisement scheme.

As a conclusion, the QoS, security and reliability services are issues that will be studied in future works once a robust and efficient multicast routing protocol is defined which is the actual purpose of this thesis.

3.2.2.5 Conclusion

As illustrated through the last paragraphs, none of the proposed protocols (to the best of our knowledge) provide a solution that meets both the robustness, the efficiency and the energy-saving requirements. On the one hand, the mesh based protocols provide the best robustness compliance but have a poor efficiency due to the duplication of data packets among the duplicate path, and to the flooding of control messages. On the other hand, the protocols that have good efficiency compliance (shared-tree, overlay...) have poor robustness characteristics.

The requirements coming from the tactical domain are strong mainly in term of bandwidth efficiency and in term of robustness. Therefore, our goal is to define a protocol that will bring together the characteristics that are needed for a protocol to be robust (alternative paths in case of link breakages) and the characteristics that make a protocol efficient (no flooding, tree structure, independence of the control overhead with respect to the number of sources...). As long as energy-saving is concerned, we will consider that if the protocol is efficient in term of data overhead (i.e. the number of nodes that participate in the data forwarding is minimum) and control overhead (few control packets need to be sent and re-transmit) over the

wireless medium, the protocol will respond to the energy-saving requirement.

3.3 Description of the Shared Tree Ad hoc Multicast Protocol

The Shared-Tree Ad hoc Multicast Protocol has been defined with the goal to be in the meantime robust AND efficient. Thus, it relies on a shared-tree structure in order to reduce at a maximum the control and data overhead. Moreover, STAMP takes advantage of the broadcast capacity of the medium to distribute the data on the tree similarly to a mesh in order to add redundancy. The node that becomes the core of the tree is the first node that joins the group.

Note that this protocol has been designed to operate as an intra-cluster multicast routing protocol. Therefore, in each cluster, a node, the clusterhead, has already been distinguished from the others and is already known by all the nodes in the clusters. Therefore, when STAMP is employed in a clustered network, the core election process is already performed by the clustering process.

Note also that in this clustered network context, the shared tree approach is the most suited to cluster topology. Indeed, we will see in the next chapter (chapter 4) that the clusterhead is a central node that needs to be part of all multicast groups to perform the inter-cluster operations. With a source-tree or with a mesh structure, it would have been less trivial to ensure that the clusterhead belongs to the structure.

In this part we describe the protocol as if it is used in a flat network. Nevertheless, we notify all the differences with the clustered-network environment use.

3.3.1 How to provide efficiency ?

In this paragraph, we describe the characteristics of STAMP that make it efficient. We remind that a protocol is said to be efficient if it minimizes the control information overhead during the construction and maintenance processes and the data duplication during the forwarding process.

3.3.1.1 General characteristics

In order to have an efficient protocol, we choose to rely on a shared-tree structure centered on a core node rather than relying on a mesh. Indeed, as long as the control information overhead is concerned, a shared-tree is the structure which minimizes the control overhead since it does not rely on any flooding for the construction phase or the maintaining phase. Concerning the duplication of data packets, a shared tree structure allows to minimize the number of duplications, and consequently to reduce the bandwidth utilization. Moreover, a shared structure is also a better response to efficiency than a source dedicated structure, since only one structure per group has to be maintained rather than as many as the number of sources. This aspect is particularly important in the context of tactical MANET, since in such networks, each member of a multicast group can also be a source for this group.

Regarding the topology maintenance mechanism, STAMP relies on a hard-state approach. The soft state approach which corresponds to the periodical refresh of

the multicast states through the use of periodical control messaging has been pushed back because of the important overhead it imply. Therefore, the multicast states in the shared-tree are updated upon detection of link breakages. The detection of link breakages is a functionality that may be implemented by the MAC layer or by the unicast routing protocol. Therefore, cross-layering functionalities would allow the information to be communicated to the multicast process. Cross-layering is an active topic of research in the field of MANET and it has proved to be a particularly relevant solution to save bandwidth and to improve the protocol operation.

Finally, the last point to consider to design an efficient protocol is the dependency with the unicast routing protocol. STAMP is partially dependent on the unicast routing protocol implemented in the MANET node which means that the unicast routing information such as the Next Hop node are needed to construct and maintain the structure but that the forwarding of the multicast packet is the responsibility of the multicast process. A tactical MANET must not only provide multicast routing capabilities but also unicast routing capabilities. Therefore, it seems obvious that a unicast routing protocol will be implemented in each tactical MANET node. Consequently, re-using the information coming from the unicast routing protocol can avoid consuming uselessly the bandwidth by duplicating some functionalities and control information exchanges. The information that can be relevant are the detection of link breakages or the Next Hop node to reach a destination. Here again, cross-layering functionalities can be employed.

To sum up, STAMP is a shared-tree-based receiver-initiated multicast protocol that relies on a hard state approach for the topology maintenance process through the re-use of information coming from the unicast routing protocol.

3.3.1.2 Tree construction

STAMP supports multicast communications thanks to a tree structure centered on a core node, i.e. a shared-tree. The tree construction initiative is given to the receiver nodes which send join messages to their upstream neighbors on the path to the core. This path is known thanks to the unicast routing protocol implemented in the MANET node. As explained previously, since a tactical MANET needs to provide not only multicast communications capability, but also unicast communications, we have chosen to base our protocol on a unicast routing protocol without dictating any conditions on the choice except that it must provide path information within a finite delay and that it must be loop free.

An on-tree node for a multicast group G stores in a multicast table its upstream neighbor on the path to the core and its downstream neighbors for the group G. When a node becomes a new member of a multicast group, it sends a join message to its next hop node on the path to the core of the group and stores it as its upstream neighbor for the multicast group. The next hop information is known from the unicast routing table. Upon reception of the join message by the next-hop node, two situations may occur as illustrated by figure 3.2.

- In the first case, the node B does not belong to the multicast tree. First, it stores the source of the join as a downstream node for the group; secondly, it sends a join_ack message to the downstream node (1); thirdly it sends a join message to its next hop node C on the path to the core (2) and finally it stores

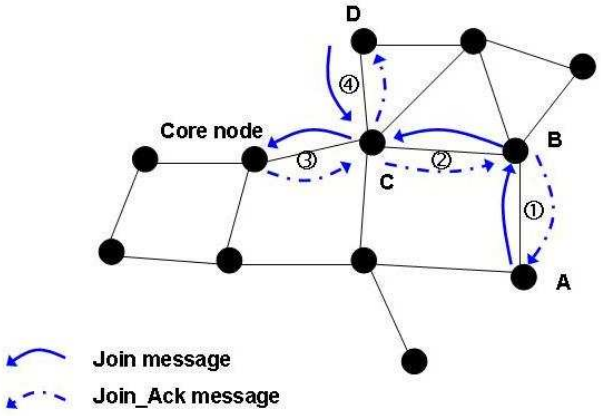

Figure 3.2 Illustration of the hop-by-hop join process

the next hop node as its upstream node for the multicast group.

- In the second case, node C already belongs to the multicast tree of the group. In this case, it stores the source of the join as a downstream neighbor for the group, then it sends back to it a join_ack message, and finally it stops the join forwarding process (4).

This forwarding process of a join message is repeated until the join reaches a node that already belongs to the tree or the core itself.

In the example of the figure 3.3(1), node A becomes a member for the multicast group. B is already an on-tree node for this multicast group. C is the next hop node to reach the core; therefore A sends a join message to C. C is not an on-tree node when it receives the join message; therefore, it must continue the joining process. C sends back to A a join_ack message and looks for its next hop to the core which is the core itself. In figure 3.3(2), node C is an on-tree node and the link $A - C$ is part of the shared-tree. Let us consider that the link from C to the core breaks while the core node is considered by C as its next hop to the core. C waits for the unicast routing protocol to find a new path to the core. This new path is through B. Therefore, C sends a join message to B which sends back a join_ack to C (figure 3.3(3)). Since B is already an on-tree node, the join process is finished and the new branch from A to B through C is constructed (figure 3.3(4)).

The tree construction process explained above follows an "hop-by-hop" approach. Indeed, in the process of construction of a new branch in a traditional shared tree protocol, the initiator of this join process waits for an acknowledgment that must be sent back to the initiator by the first node that already belongs to the tree on the path from the initiator to the core. During this process of joining the tree, if a link in the new branch breaks, the node that has initiated the construction of this new branch must re-start the process since it does not receive the acknowledgment it is waiting for. If this event is a priori rare in a wired environment, it may happen often in a MANET. The process of reconstruction followed by traditional shared-tree-based multicast routing protocols mentioned previously generates not only a latency but also an additional overhead. Thus, to solve this problematic, the process of creation of a new branch taken by STAMP follows an "hop-by-hop" approach, which means

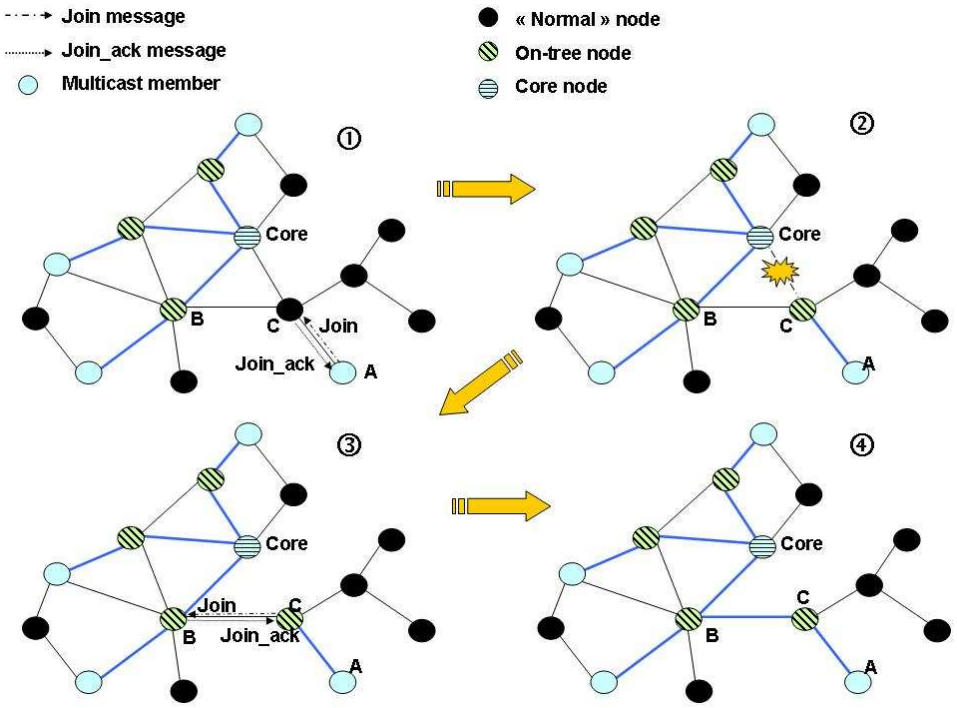

Figure 3.3 Illustration of the tree construction process

that the responsibility for constructing the new branch is passed from node to node on the path from the new receiver to the core. By sending a join message to its upstream neighbor, the receiver makes the upstream neighbor responsible for the new branch establishment. Consequently, the reception of a join message means that the node is requested by its downstream neighbor to be part of the tree, and that it is now up to the node to construct the branch of the multicast tree. That is why in the second phase of the joining process described above, the next hop node sends back a join_ack message even if it is not an on-tree node. As soon as the source of the join receives the join_ack message from its upstream node, it becomes a tree member and thus it is able to forward any multicast datagram it may receive. It is also able to receive any multicast traffic for the multicast group since it is now a member. Thereby, the tree construction responsibility goes hop-by-hop from the receiver to the core node until reaching a node that is already on the tree or the core.

Hence, when a link breaks during this joining process, two situations may occur:

- If the link that breaks is upstream from the last node that has received the join message, it is the case of a classical link breakage that is described in the following of this dissertation.

- If the link that breaks is downstream from the last node that has received the join message, it is up to this last node to find a new path to the core.

The process of joining does not have to be re-started from the initiator.

A special case may happen in case the route to the core exists when the join is initiated by the receiver and is lost when the join goes hop-by-hop to the core. In this

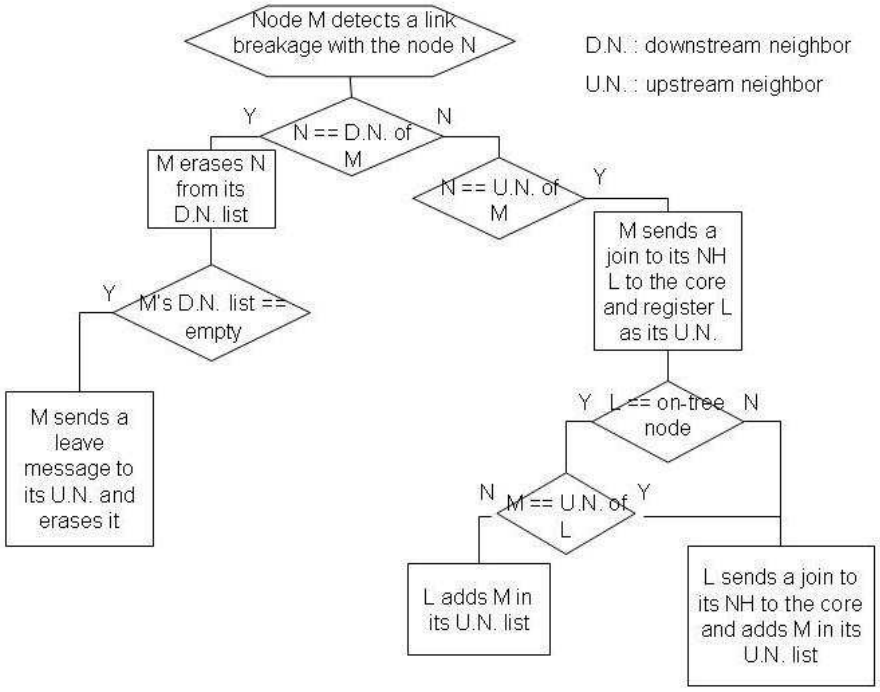

Figure 3.4 State Diagram of the processing of a link breakage

case, it is up to the node that does not have a route anymore and that receives the join, to send back to the initiator of the join process (the new member) a join_nack message. The join_nack follows the tree branch that was in creation. When receiving a join_nack message, a node erases the states that it has set up.

One can argue that the hop-by-hop construction may introduce suboptimal branches (i.e. branches not corresponding to shortest paths between the core and a receiver) if a topology change occurs while the join message travels to the core. This issue is solved by the tree maintenance mechanism.

Note that in STAMP, a source of a multicast group does not need to join the multicast delivery structure to send a datagram to the group.

3.3.1.3 Tree maintenance

The maintenance process ensures that each node remains connected to the tree. This process is very important in MANET environments where the mobility of nodes, as well as wireless links, can cause frequent link breakages and topology changes. With the efficiency compliance in mind, we choose a hard state approach for the tree maintenance process. Hence, link breakages must be detected by the MAC layer or thanks to the unicast routing protocol.

When a node that belongs to a multicast tree is informed that it has lost the connectivity to a neighbor node (see fig.3.4):

- If the lost node is one of its downstream nodes for a multicast group, it erases the lost node from its downstream node list. If its downstream node list be-

comes empty and if the node is not a member for the group, it must leave the tree. Therefore, it sends a leave message to its upstream neighbor.

- If the lost node is its upstream node for a multicast group, it must repair the branch. Thus, it sends a join message to its next hop node on the path to the core. At this point, two situations may occur.
  - First, the upstream node already belongs to the tree, and the sender of the join is not its upstream node. In this case it simply adds it to its downstream list.
  - The second situation is when the upstream node does not belong to the tree, or when it belongs to the tree but the node sending the join is its current upstream node. In this case, the upstream node must send a join message to its next hop node on the path to the core, registers it as its new upstream node and adds the sender of the join in its downstream node list.

With such an algorithm (especially due to the hop-by-hop tree construction), each branch of a multicast tree is not necessarily formed by the shortest path between a receiver and the core. To achieve such a property, the on-tree nodes periodically check their routing table to verify whether their upstream nodes are still their next hop to the core. If the negative, a join message is sent to the next hop node on the path to the core in order to set up a new branch. When a datagram is received on this new set up branch, the old branch can be erased (i.e. a leave message is sent to the old upstream node).

Let us consider an example of tree reconstruction (figure 3.5) illustrating some of the processes described previously. The multicast shared-tree has been constructed following the processes described in the previous paragraph. Let us consider that the link between node B and the core breaks (figure 3.5(1)). The upstream node of the link breakage is the core. It removes B from its downstream list. It is a member of the multicast group and its downstream node list is not empty. Therefore, it does not have to do anything else. B is the downstream node of the link breakage. It has to find a new path to the core. In this new path, the next hop node to the core is A. Thus, B sends join message to A which sends back a join_ack message to B. A is already an on-tree node, therefore A just adds B to its downstream list (figure 3.5(2)). When checking if its upstream node corresponds to its next hop to the core, C realizes that this condition is not verified. Therefore, C sends a leave message to B and sends a join message to D which is its next hop to the core. D sends back a join_ack message to C (figure 3.5(3)). D is not an on-tree node therefore it has now the responsibility for creating the branch from C to the core. D sends a join message to the core which responds by a join_ack. At the end of this process, the tree is repaired and is made of the shortest paths from the members to the core.

3.3.1.4 Leaving a multicast group

When a node wants to leave a multicast group, it first checks if its downstream node list is empty. If the negative, it must remain an on-tree node. It the positive, it can leave the tree. Therefore, it sends an explicit leave message to its upstream node for this multicast group. When a node receives a leave message, it simply erases

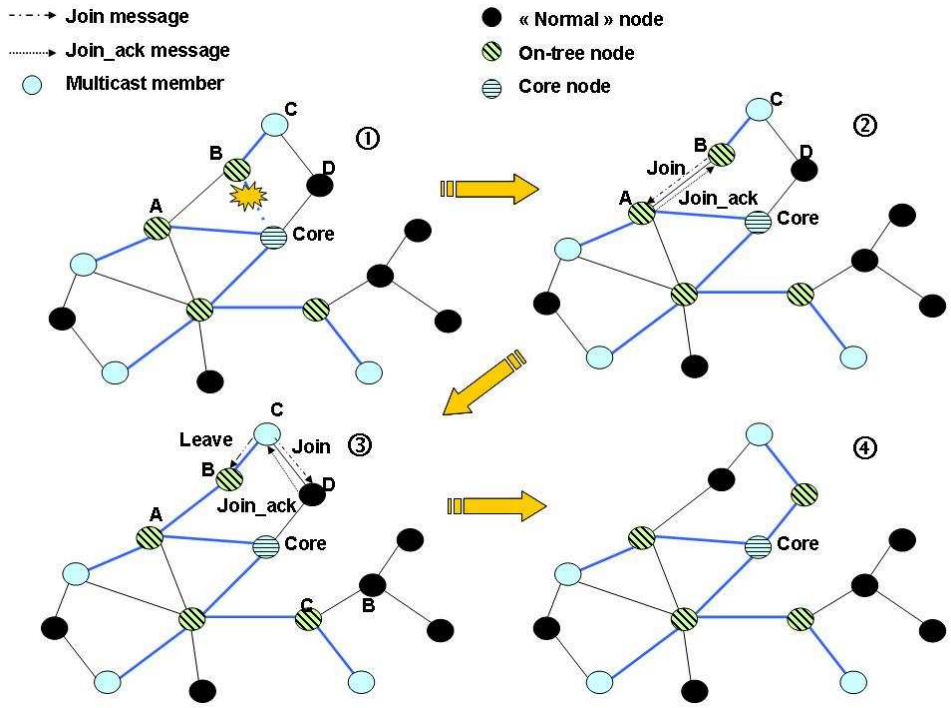

Figure 3.5 Illustration of the tree maintenance process

the source node of this leave message from its downstream node list. When the downstream node list for a multicast group becomes empty and if the node is not a member of the group, the node does not need to be part of the tree anymore, therefore, it sends a leave message to its upstream node.

3.3.1.5 Core election

The algorithm to elect the core of a multicast group is rather simple. When a node becomes a member of a multicast group, it checks whether it already knows a core for this group. If the negative, it elects itself as the core for this group. Consequently, it broadcasts a core announcement message on the network with its node id as the core id. When receiving a core announcement message, if it is not a duplicate message, a node stores the association "Multicast group/Core id" in a table. If several nodes become members in the meantime, it is the one with the highest id that becomes the core. Thereby, when a core node receives a core announcement from a node with a higher id, it releases its core node status and becomes an on-tree node.

Note that if the protocol is implemented in a clustered network, this process does not have to be done, since the core node corresponds to the clusterhead of the cluster. Each node in the cluster knows the address of its clusterhead.

3.3.2 How to provide robustness?

Tree-based protocols are known to be poorly robust because they cannot provide alternative paths to continue forwarding the multicast data when a link of the tree

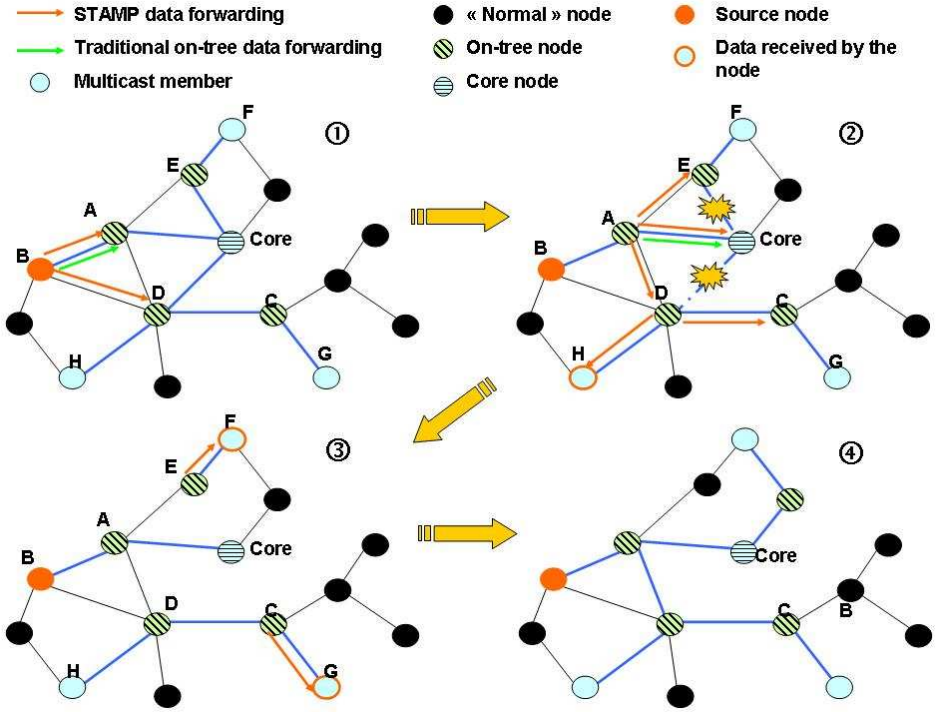

Figure 3.6 Illustration of the data forwarding process

breaks. In order to make our protocol more robust than a traditional shared-tree, the way packets are forwarded on the tree has been modified.

In traditional tree-based protocol, data packet follows the branches of the tree. This means that a multicast data packet can be accepted by a node for forwarding only if it has been received either from an upstream or a downstream node. This is what we can call the "forwarding rule". Following this process, the fact that a packet can not be forwarded any more when a link of the tree breaks seems inherent to the tree topology. Nevertheless, this forwarding process does not take into account the broadcast capacity of the wireless medium. Indeed, when a node on the tree forwards multicast data packet, this packet can be heard by all its neighbors. Therefore, if one of its neighbors is an on-tree node of another branch of the tree, it can be an opportunity to improve the robustness of the protocol to allow that this neighbor accepts the packet for forwarding. Finally, the only think to do is to modify the acceptance rule of a multicast data packet to introduce redundancy in the tree structure. Thus, the "forwarding rule" becomes that an on-tree node always accept a data packet for forwarding from any neighbors if it is the first time this packet is received by the node. At MAC layer, a multicast data packet must be operated similarly to a 1-hop broadcast packet.

Let us illustrate this process by an example (figure 3.6). In this example, B is a source for the multicast group. Nodes F, G, H and the core are members. Nodes A, C, D and E are on-tree nodes. The orange arrows illustrate the STAMP forwarding process and the purple arrows illustrate the tradition tree forwarding. The source starts sending its first packet. Since the source is also an on-tree node, the packet is sent to its upstream node according to the traditional process. With STAMP,

both A and D which are neighbors of B and on-tree nodes receive the multicast data packets sent (figure 3.6(1)) and forward them to their neighbors. Node A, the core and C receive the multicast packets. With the traditional approach, node A only forwards the data packets to its upstream node which is the core. Let us assume that the links between node E and the core and between node D and the core break (figure 3.6(2)). With the traditional approach, data packets are lost until the tree is repaired since the core becomes isolated from the remainder of the tree. With the approach taken by STAMP, since nodes C, and E have already received the multicast data packet, they can go on forwarding it (figure 3.6(3)). Finally, with STAMP approach even if two links break, all members receive the multicast data packets. Moreover, the latency can be reduced. For example, node H receives the data packet in two hops (B-D-H), whereas with a traditional forwarding, it would have received the packet in four hops (B-A-Core-D-H). Figure 3.6(4) presents the updated tree after the maintaining process.

It should be noted that this forwarding process does not increase the forwarding overhead compared to a tree approach since, only on-tree nodes forward the data once, exactly the same as in a traditional forwarding.

If the source of a multicast group is not an on-tree node, it sends its multicast packets in unicast to its next hop node to the core. This process is repeated until the packets reach the core or an on-tree node. If the source node does not know the core for the group, it means that no node in the network is member for this group, otherwise it would have received the core announcement message. Therefore it does not have to send its data.

3.4 Performance evaluation of STAMP

In the previous paragraph, we present our proposal: Shared-Tree Ad hoc Multicast Protocol emphasizing on the characteristics that make it efficient and robust. In this part, we evaluate the performance of STAMP through discrete event simulations. The goals of this performance evaluation is to verify that STAMP is really an efficient and robust multicast protocol and to compare it with a mesh based multicast protocol which is the category of multicast protocols currently known to reach the best compromise between robustness and efficiency. The remainder of this section is organized as followed. In a first part we present the framework of this performance evaluation. Then, the metrics that are observed to analyze and compare the protocols are given. In a third part, the scenarios we work on are described and finally the results are commented in the last part.

3.4.1 Framework

We perform discrete event simulations thanks to the OPNET Modeler 11.5 simulator [95]. We choose to compare STAMP to ODMRP which is the most representative of the mesh-based multicast routing protocols for MANET. The model of ODMRP has been provided by a third party [1] and slightly modified to be compliant with the

[1] We thank MAJ Fernando J. Maymi, Assistant Professor in the Dept. of Electrical Eng. & Computer Science of the U.S. Military Academy, West Point for providing his model of ODMRP.

Table 3.2 ODMRP simulation parameters values

Parameter	Value
Route Refresh Interval	3 seconds
Forwarding Group Timeout	9 seconds

Table 3.3 OLSR simulation parameters values

Parameter	Value
Hello Message Interval	2 seconds
Topology Control Message Interval	4 seconds
Neighbor Hold Time	6 seconds
Topology Hold Time	12 seconds

IETF Internet Draft [136]. The parameters of ODMRP are given in table 3.2.

Our network is composed of 50 mobile nodes with a propagation radio range of 250m randomly placed within a 1000m*1000m area. The Optimized Link State Routing (OLSR [28]) protocol is employed as the underlying unicast routing protocol for STAMP with the parameters defined in table 3.3. No unicast routing protocol is employed in the simulation with ODMRP. The control overhead generated by OLSR is expected to influence the performance of STAMP. For the MAC layer, the 802.11 WLAN using Distributed Coordination Function is used with a channel capacity of 2Mbits/s. The Direct Sequence Spread Spectrum is employed as the modulation technique. The buffer size of the MAC layer is of 256 Kbits. Each node moves randomly according to the Random Waypoint model [16, 86] with no pause time. At the beginning of the simulation, each node selects a random destination in the area and moves to this destination at a speed defined by a parameter of the mobility model. Upon reaching this destination, it chooses randomly another destination and moves to this destination without waiting. The speed can be chosen randomly between two boundaries for each segment or may be fixed at the beginning of the simulation. The multicast traffic is a Constant Bit Rate (CBR) traffic where the size of a packet is 512 bytes. For each scenario, multiple runs of 500s with different seeds were run. These simulation environment characteristics are similar to those considered in the multicast routing protocol comparison paper [78] proposed by the ODMRP developers.

3.4.2 Metrics observed

We follow the suggestions of the IETF MANET working group [30] for evaluating routing or multicasting protocol. The following metrics are chosen:

The Packet Delivery Ratio (PDR): the number of the received packets divided by the number of packets expected to be received, where the number of packets expected to be received is the number of data packets sent by the sources times the number of receivers.

The Data Packet Overhead (DPO) i.e. the number of data packets transmitted on the network per data packet delivered, it measures the number of individual copy of data packets transmitted on the whole network.

The Control Bits Overhead (CBO) i.e. the number of control bits transmitted per data bit delivered, it measures the control overhead needed to install and maintain the tree structure with respect to the data delivered.

The Total Packet Overhead (TPO) i.e. the total number of packets (control and data) transmitted on the network per data packet delivered, it measures the total number of packets to be transmitted on the network (control and data) to achieve the transmission of the datagrams.
The average End-to-End delay from source to destination nodes, it measures the average delay from sources to receivers. This time may depend on the number of hops as well as re-transmissions of packets due to congestion.

The Packet Delivery Ratio is an important metric since it gives the loss rate that the transport protocols have to face to, which affects the maximum throughput of the network. This metric is used to characterize the completeness and the correctness of the protocol. In case of mobility, this metric gives a relevant information about the robustness of the protocol.

The number of time a data packet should be transmitted to reach the destination allows to measure the capacity of a protocol in using the bandwidth efficiently. The Data Packet Overhead is used to evaluate this capability. In networks where the traffic is dense compared to networks where the traffic is sparse, this metric is all the more critical.

For the control overhead, we choose to rely on the number of bits rather than on the number of packets sent. Indeed, sending a short control message does not have the same impact on the bandwidth occupation than sending a long control message. This aspect is not taken into account when relying on the number of packets sent rather than on the number of bits. The number of control packets sent appears in the Total Packet Overhead. Indeed, the difference between the Total Packet Overhead and the Data Packet Overhead gives the Control Packet Overhead. The control overhead allows to measure the scalability of the protocol, the degree to which it will function in congested or low-bandwidth networks and its efficiency in consuming the node resource (node battery power) and the network bandwidth.

3.4.3 Simulation scenarios

Protocol performance are observed in several network configurations where some parameters evolve in order to measure the impact of these parameters on the protocol. Five sets of experiments are executed to study the effect of the node mobility, the number of source nodes, the number of multicast members, the traffic load and the density of the network. The number of multicast groups is set to 1. In each set of experiments, all the metrics described previously are observed. For each set of experiments, only one parameter changes while the others are held constant to a medium value so that we can study the impact of each parameter independently (see Table 3.4). For each set of experiments, we derive a number of scenarios. For example, in the node mobility experiment, the node mobility evolves from 0 to 20 m/s leading to 6 scenarios while the number of sources, the number of multicast members, the traffic load and the density of the network remain constant. For each scenario, a number of runs are performed with different seeds. The seed is used to generate random numbers (employed in the Random Waypoint mobility) and to choose randomly the member nodes and the sender nodes. The observed metrics are then averaged on these different runs. A confidence interval can then be computed.

Multicast member nodes are chosen randomly among the 50 nodes and sources

Table 3.4 Overview of the simulation scenario parameters

	Speed (m/s)	Nb of sources	Nb of multicast members	Traffic Load (pkt/s)	Area size (m)
Experiment 1	0, 2, 5, 10, 15, 20	5	20	10	1000*1000
Experiment 2	5	1, 2, 5, 10, 20	20	10	1000*1000
Experiment 3	5	5	5, 10,20 30, 40	10	1000*1000
Experiment 4	5	5	20	1, 2, 5, 10, 25, 50	1000*1000
Experiment 6	5	5	20	10	800 to 1600

nodes are chosen randomly among multicast members. The traffic load corresponds to a global network load equally distributed among sources nodes. For the density experiment, since the number of nodes in the network and the propagation range are fixed respectively to 50 and 250m, it is the size of the network domain that is modified in each scenario. The metrics values are collected from the time 50s i.e. the first 10% of the simulation time are excluded.

3.4.4 Simulation results and analysis

In this section, we present and analyze the results of the performance evaluation scenarios based on the metrics defined previously. We describe the influence of various parameters on the operation of the protocol i.e. the mobility, the number of sources per group, the number of multicast members per group, the traffic load and the network density. In some of the presented figures, the concavity of the curve changes. If we had traced the confidence intervals, such evolution would not have appeared. Nevertheless, we do not have represented the confidence intervals because we do not want to overload the figures and also because there are good but not excellent. This last point is due to the fact that the simulation time is long for each run (more than one hour) and therefore, "only ten" runs of each scenario have been simulated. If we had have time to perform at least twice more runs for each scenario, the confidence intervals would have been better.

3.4.4.1 Influence of the mobility

In this part, we present the results of the first experiment to study the influence of the mobility on the operating of the protocol. Six values of mobility are considered: 0, 2, 5, 10, 15 and 20 m/s. There is one multicast group composed of 20 members and 5 sources. We remind that nodes move following the Random Waypoint mobility model. Figures 3.7, 3.8, 3.9, 3.10, and 3.11 present the different metrics as a function of the node speed.

The figure 3.7 shows that STAMP achieves a high Packet Delivery Ratio (PDR) for a tree based protocol. Usually, in the same conditions, tree based protocols PDR falls rapidly to 50% [78, 125]. Nevertheless, it is still under the ODMRP one, especially when mobility increases. This is due to the fact that the tree created by STAMP is less dense than the mesh created by ODMRP. Therefore, when a route breaks, STAMP must often wait for OLSR to provide a new route which may take time since OLSR re-computes its route every 4 seconds. During this time, some data packets may be lost if no alternative route exists which may happen more often in STAMP than in ODMRP. Moreover, OLSR generates an overhead that may fill the

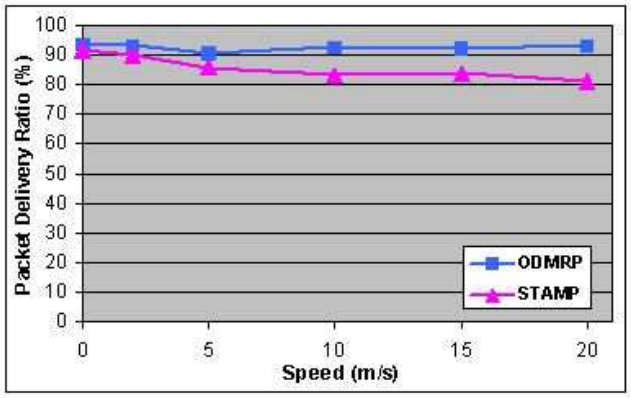

Figure 3.7 Packet Delivery Ratio Vs. Mobility

transmission buffers and may make packets be dropped.

The figure 3.8 shows that STAMP generates much fewer signaling overhead than ODMRP. Indeed, ODMRP uses periodic source flooding (every 3 seconds). It is to notice that the control overhead is increasing in STAMP when mobility increases which is due to the hard state approach of tree maintenance but remain very low. As expected and illustrated by figure 3.9, data overhead in STAMP is inferior to data overhead in ODMRP since the virtual mesh used for data forwarding in STAMP is less dense than the Forwarding Group of ODMRP. In average, ODMRP sends one copy more than STAMP of each data packet. With the Total Packet Overhead figure (fig. 3.10), we can deduce that STAMP generates very few control messages whereas ODMRP generates more control messages. Indeed, the difference between the Total Packet Overhead and the Data Packet Overhead curves gives the amount of control messages. Therefore, it is not because ODMRP sends longer control messages that the control overhead of ODMRP is higher but rather due to the fact that ODMRP generates more control messages. Moreover, the fact that the Data Packet Overhead remains stable even when the mobility is high proves that STAMP operates correctly even in case of important stress. No routing loop appears, and the structure is stable.

The figure 3.11 presenting the End-to-End delay shows that even if the tree created by STAMP does not ensure shortest paths between sources and destinations (it is a shared-tree), the forwarding process using a "virtual" mesh allows to achieve the same end to end delay than ODMRP which uses shortest paths between each source and each destination. Moreover, the fact that the delay is stable even when the mobility increases confirms that the structures are well maintained and that no routing loop is formed.

This first set of experiments proves that STAMP is robust with respect to mobility. Indeed, it achieves high delivery ratio meaning that the structure is well maintained. Moreover, the fact that the Data Packet Overhead is stable proves that the structure does not integrate nor eliminates useless nodes which means that the protocol correctly maintains the tree. Moreover, in term of efficiency, STAMP is better that ODMRP since it generates much fewer control and data overhead than its counterpart. The figure 3.12 illustrates the benefit in efficiency achieved by STAMP compared to the little loss in term of robustness. Indeed, this figure gives the difference in percentage between the data overhead (in blue), the packet delivery ratio

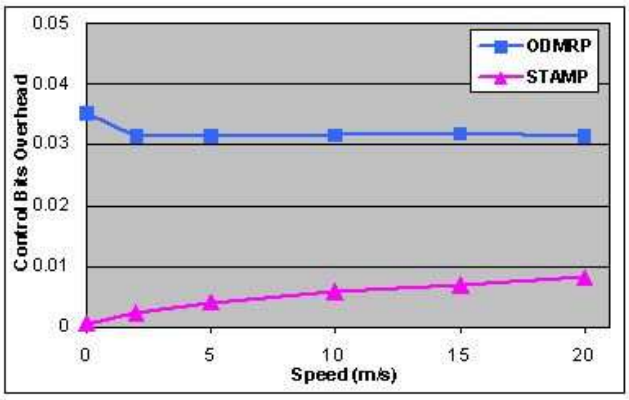

Figure 3.8 Control Bits Overhead Vs. Mobility

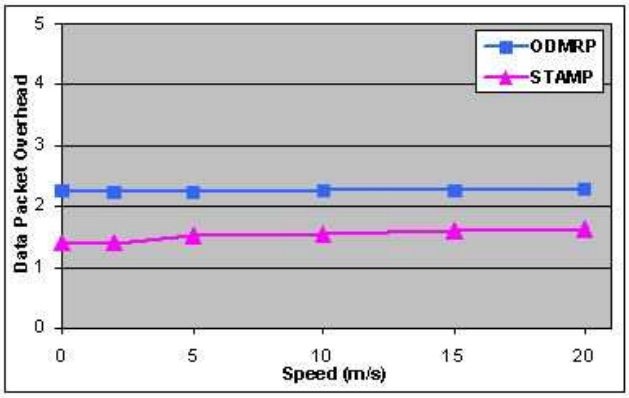

Figure 3.9 Data Packet Overhead Vs. Mobility

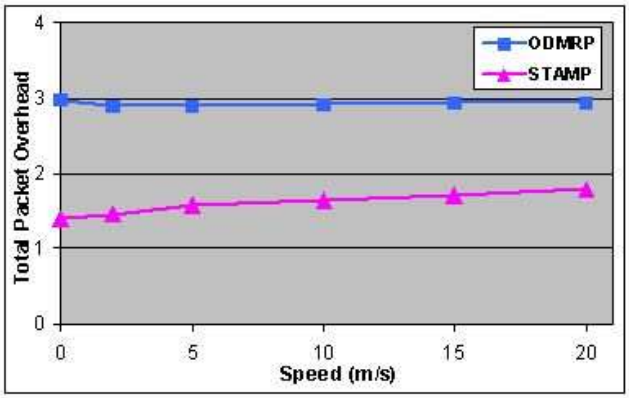

Figure 3.10 Total Packet Overhead Vs. Mobility

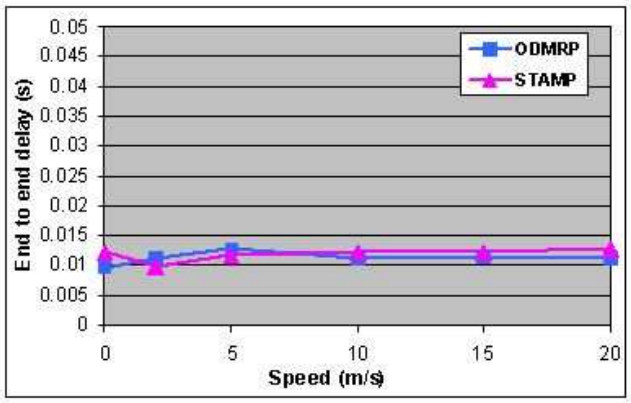

Figure 3.11 End To End Delay Vs. Mobility

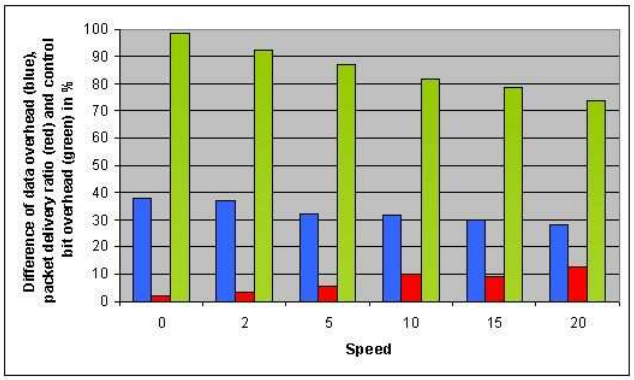

Figure 3.12 Comparison of the data overhead, the packet delivery ratio and the control bits overhead Vs. Mobility

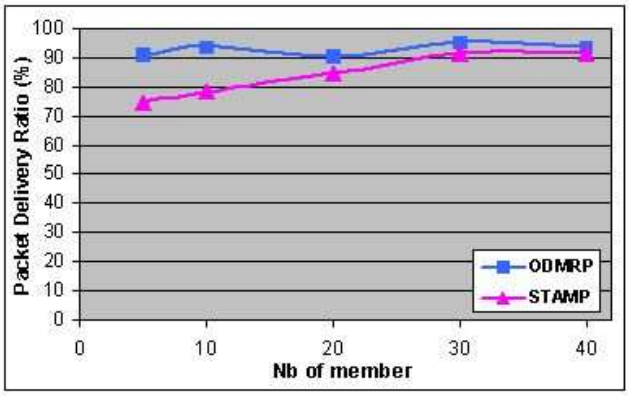

Figure 3.13 Packet Delivery Ratio Vs. Number of Multicast Members

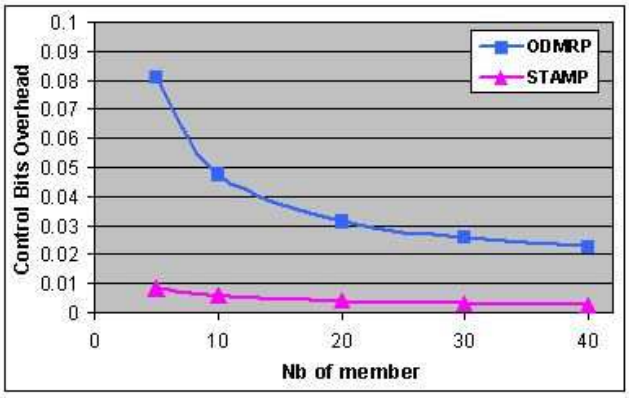

Figure 3.14 Control Bits Overhead Vs. Number of Multicast Members

(in red) and the control bits overhead (in green) of ODMRP and STAMP. For example, for a node speed of 2m/s, STAMP produces 98% less control bits overhead than ODMRP, 36% less data overhead and achieves only 3% less delivery ratio than ODMRP.

3.4.4.2 Influence of the number of members per group

In this part, we present the results of the second experiment to study the influence of the number of multicast members on the operation of the protocol. Five values are considered: 5, 10, 20, 30 and 40 nodes. There is one multicast group composed of 5 sources. Each node moves following the Random Waypoint mobility model at a speed of 5m/s. Figures 3.13, 3.14, 3.15 and 3.16 present the different metrics as a function of the number of multicast members.

When the number of members is low, STAMP suffers from lower delivery ratio than ODMRP (figure 3.13). The multicast tree is very sparse and therefore, the forwarding process cannot rely on much virtual redundancy (the fact that two neighbor nodes may belong to two different branches of the tree). Indeed, the probability to have neighbors that belong to two different branches is low. Therefore, when a branch breaks, we must wait for OLSR to provide a new path to the core leading to

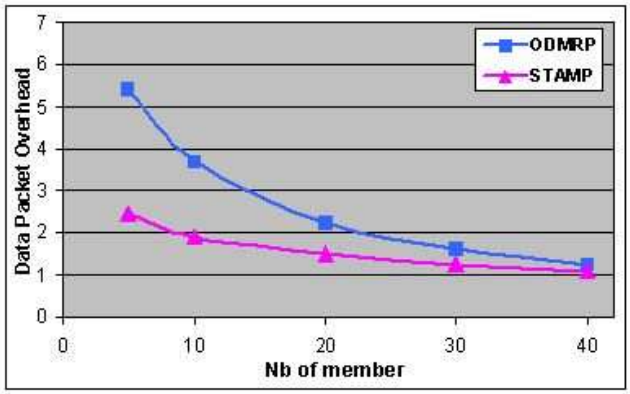

Figure 3.15 Data Packet Overhead Vs. Number of Multicast Members

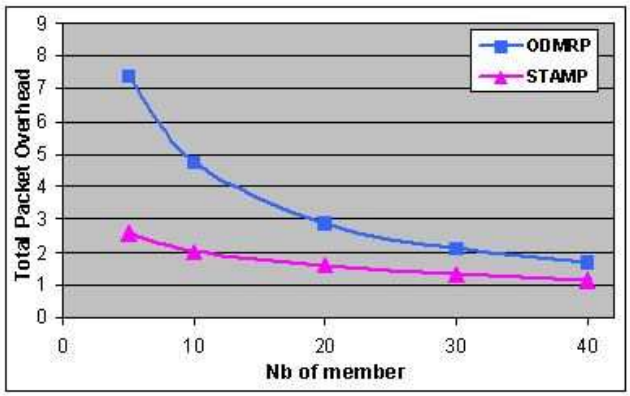

Figure 3.16 Total Packet Overhead Vs. Number of Multicast Members

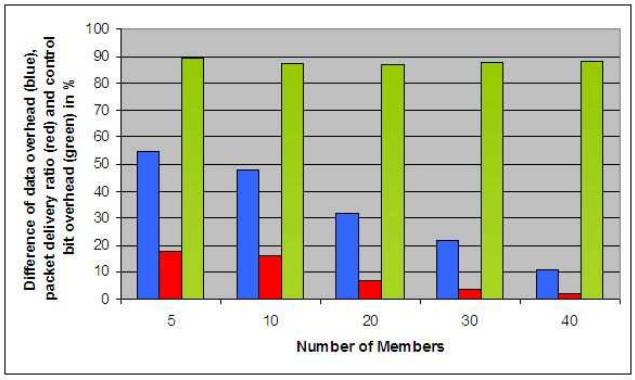

Figure 3.17 Comparison of the data overhead, the packet delivery ratio and the control bits overhead Vs. Number of Multicast Members

packet losses. When the number of multicast members increases, the Packet Delivery Ratios of the two protocols converge. This is due to the fact that the tree created by STAMP and the mesh created by ODMRP tend to "merge". This is confirmed by the figure 3.15 on the data packet overhead where the curves converge when the number of multicast members increases. The Data Packet Overhead decreases when the number of members increases since a single copy of a data packet can reach more receivers. The Data Packet Overhead converges to 1 which should be reached when all nodes are members.

The Control Bit Overhead (figure 3.14) is still inferior with STAMP even when the number of members increases which is generally a configuration that is preferable for a mesh. Indeed, the flooding of control messages employed by ODMRP is less penalizing when almost all nodes are members. In ODMRP, when the number of members is low, the Total Packet Overhead is very important. Indeed, almost seven packet transmissions (control and data) are needed to deliver a packet to a multicast member. The figure 3.16 confirms that the control overhead of ODMRP is due to a larger number of control packets rather than larger control packets.

As illustrated by the figure 3.17, the benefits in term of data overhead and control overhead are far more important that the loss in term of Packet Delivery Ratio. For example, STAMP produces 31% less data overhead than ODMRP when the size of the multicast group is 20, STAMP packet delivery ratio is 7% inferior than the ODMRP one and the control overhead is 87% inferior. This set of experiments confirms that STAMP provides a very good efficiency and maintains a good delivery ratio.

3.4.4.3 Influence of the number of sources per group

In this part, we present the results of the third experiment to study the influence of the number of multicast sources on the operation of the protocol. Five values of number of sources are considered: 1, 2, 5, 10 and 20 nodes. There is one multicast group composed of 20 members. Each node moves following the Random Waypoint mobility model at a speed of 5m/s. Figures 3.18, 3.19, 3.20, 3.21 and 3.22 present the different metrics as a function of the number of multicast sources.

STAMP and ODMRP present the same evolution of the Packet Delivery Ratio

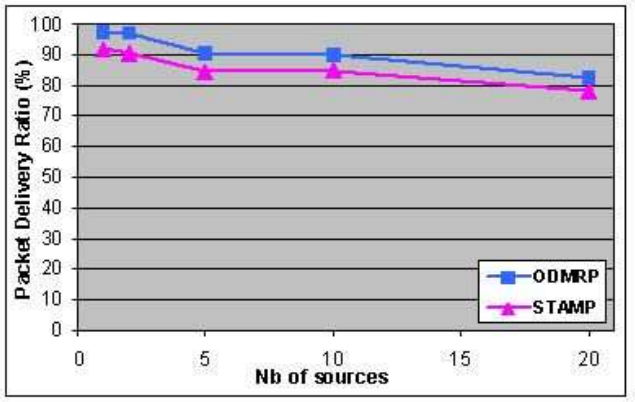

Figure 3.18 Packet Delivery Ratio Vs. Number of Sources

when the number of sources increases (figure 3.18). This fact proves that this decline is not due to collisions with control overhead (otherwise STAMP should not face this fall) but rather to collision of the different data packets coming from the different sources within the structure. Indeed, in a structure where there is only one source, all data packets follow the same path and therefore cannot hardly experience collisions among them since there are sent and then treated sequentially by all nodes in the structure. In a structure where there are multiple sources, packets coming from one side of the structure may collision with packets sending by another source at the other side of the structure. Therefore, this decline in the packet delivery ratio is mainly due to the traffic pattern and the fact that all the sources transmit packets at the same time. Finally this figure does not allow to draw conclusions on the influence of the number of sources in the protocol operation.

Figure 3.19 and 3.20 show that the control and data overheads in STAMP are not influenced by the number of sources. This is mainly due to the fact that STAMP creates a single receiver-initiated tree independently from the number of multicast sources. At the opposite, in ODMRP, each source periodically floods join messages to construct and maintain the mesh. This leads to the control overhead "explosion" that can be observed on figure 3.19 and to the increase in data overhead that is observed in figure 3.20. Indeed, the mesh is denser as the number of sources increases.

The End-to-End delay (figure 3.22) for ODMRP increases because of the important overhead (control and data) that imposes to each node to buffer data before forwarding whereas in STAMP the End-to-End delay is quite stable.

Here again, the experiment confirms the high efficiency of our protocol.

3.4.4.4 Influence of the traffic load

In this part, we present the results of the fourth experiment to study the influence of the traffic load on the operation of the protocol. Six values of the load are considered: 1, 2, 5, 10 , 25 and 50 packets per second. Since each packet is of size 512 bytes, the different traffic loads considered in kbits/s are 4, 8, 20, 40, 102 and 404. There is one multicast group composed of 20 members and 5 sources. Each node moves following the Random Waypoint mobility model at a speed of 5m/s. Figures 3.24, 3.25, 3.26, 3.27 and 3.28 present the different metrics as a function of the traffic load.

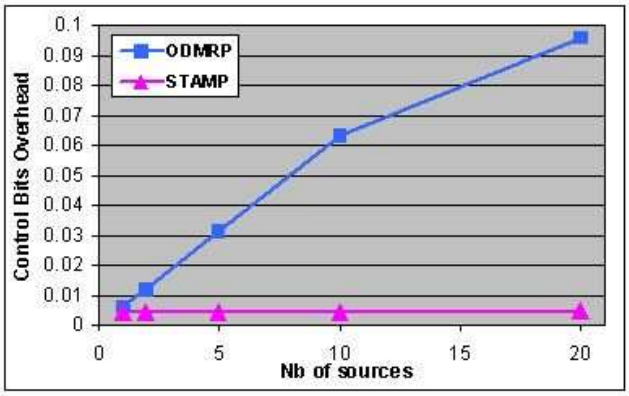

Figure 3.19 Control Bits Overhead Vs. Number of Sources

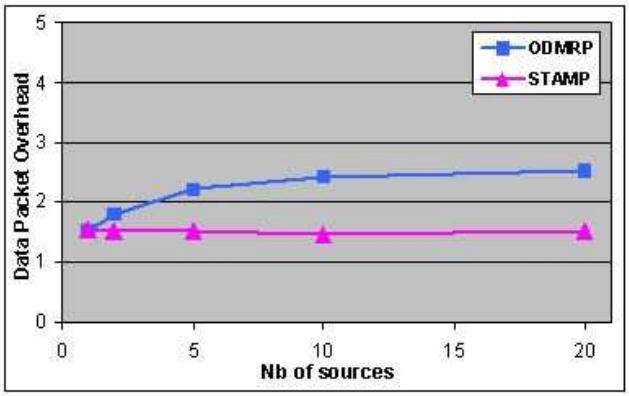

Figure 3.20 Data Packet Overhead Vs. Number of Sources

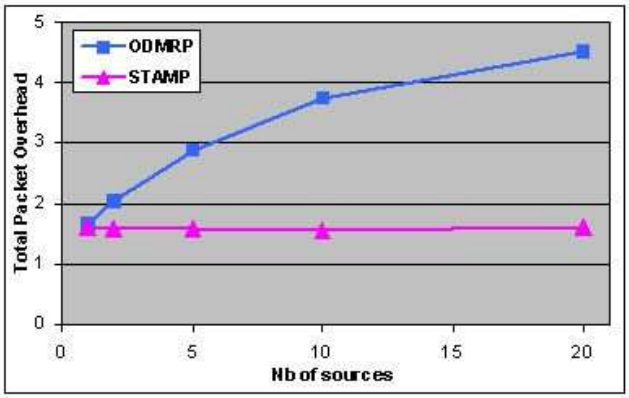

Figure 3.21 Total Packet Overhead Vs. Number of Sources

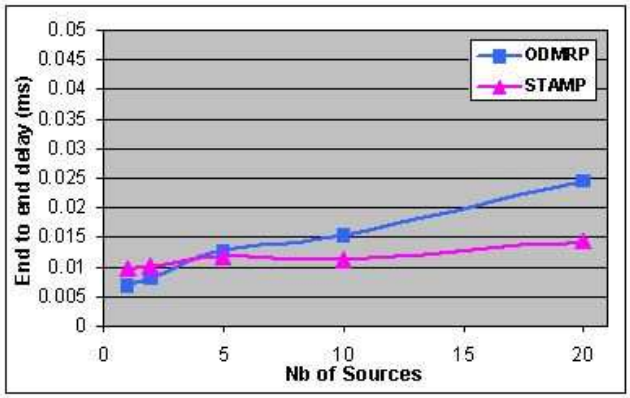

Figure 3.22 End To End Delay Vs. Number of Sources

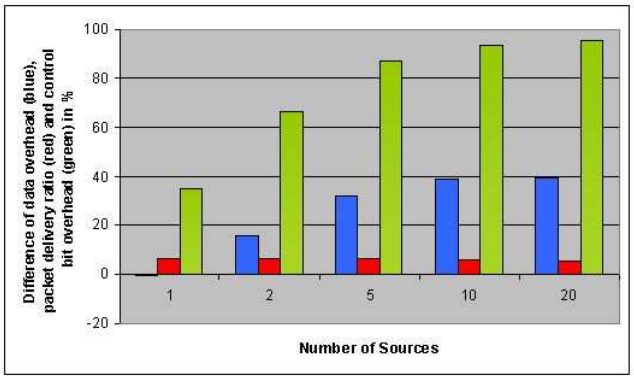

Figure 3.23 Comparison of the data overhead, the packet delivery ratio and the control bits overhead Vs. Number of Multicast Sources

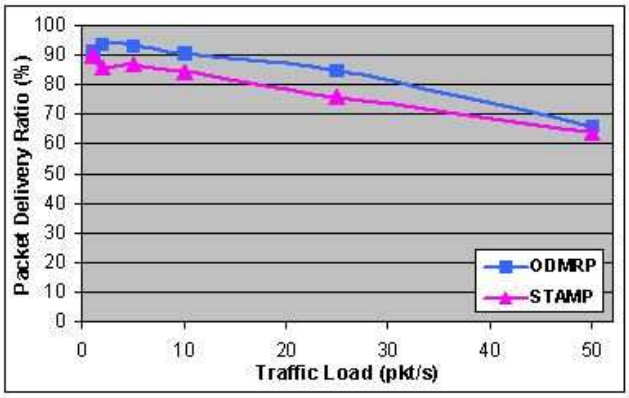

Figure 3.24 Packet Delivery Ratio Vs. Traffic Load

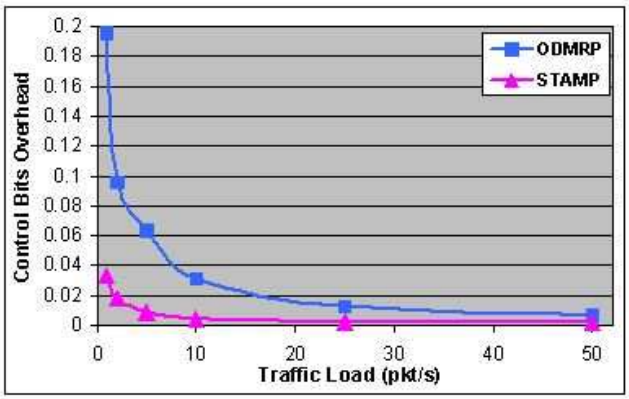

Figure 3.25 Control Bits Overhead Vs. Traffic Load

When the traffic load increases, the Packet Delivery Ratio declines (figure 3.24). Indeed, the data traffic generates collision on the support, congestion on the transmission buffers and finally packets are dropped before being transmitted. This is confirmed by the other figures that show that the traffic load does not influence the operation of the protocol.

The figure 3.25 presents the Control Bit Overhead as a function of the traffic load. As expected, the control bits overhead decreases for both protocols showing that the traffic load as no influence on the control overhead generated by the protocol. Indeed, we can see that the control overhead is conversely proportional to the traffic load. Nevertheless, STAMP still presents a better Control Bit Overhead than ODMRP. The figure 3.26 presenting the Data Packet Overhead shows that the protocol operation is not influenced by the traffic load since the number of data packets that is sent in the network is stable with respect to the traffic load. This proves that STAMP provides a good resistance to traffic load and to collisions at the MAC layer.

3.4.4.5 Influence of the network density

In this part, we present the results of the last experiment to study the influence of the network density on the operation of the protocol. Five values of the density

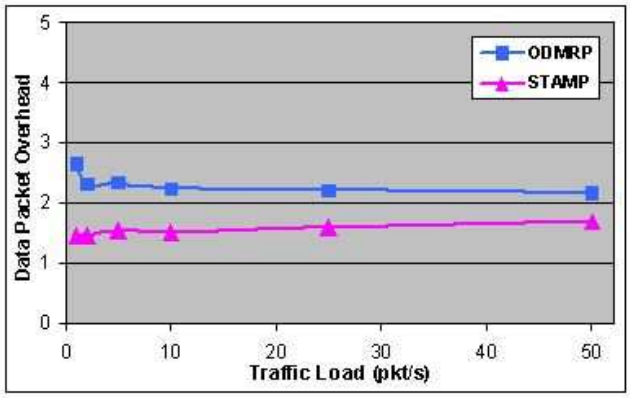

Figure 3.26 Data Packet Overhead Vs. Traffic Load

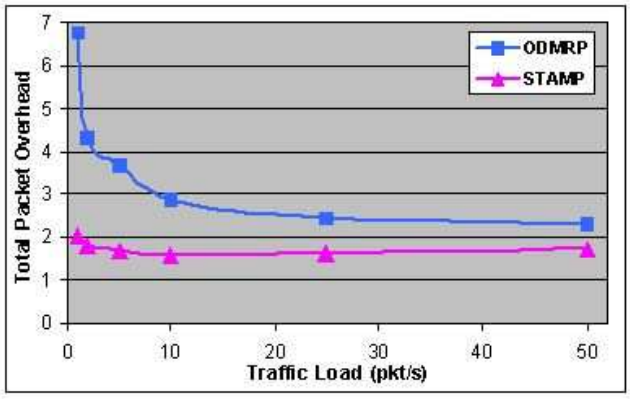

Figure 3.27 Total Packet Overhead Vs. Traffic Load

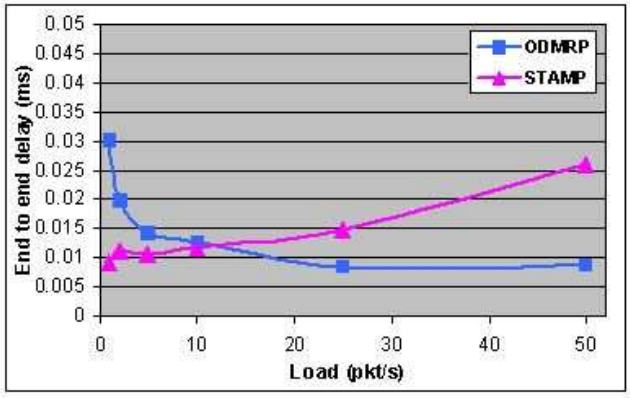

Figure 3.28 End To End Delay Vs. Traffic Load

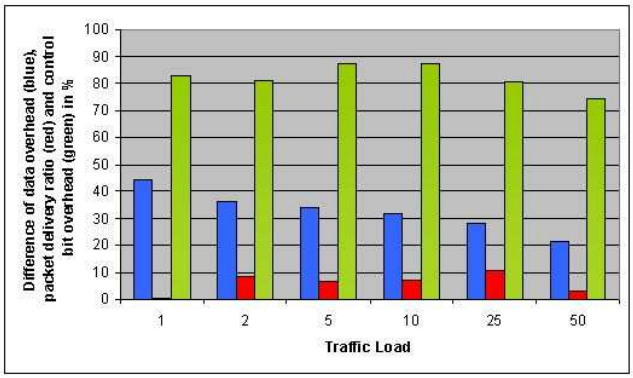

Figure 3.29 Comparison of the data overhead, the packet delivery ratio and the control bits overhead Vs. Traffic Load

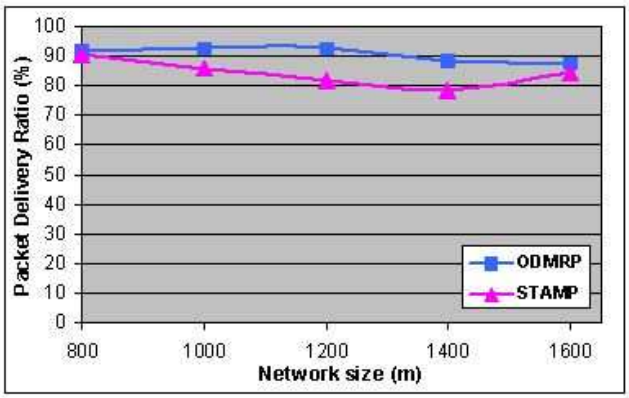

Figure 3.30 Packet Delivery Ratio Vs. Network Density

or connectivity are considered. Since the number of nodes and the propagation distance are fixed, we choose to vary the network size to make the density evolve. The network is represented by a square where the side of the square takes the values 800, 1000, 1200, 1400 and 1600 m. In this set of experiments, there is one multicast group composed of 20 members and 5 sources. Each node moves following the Random Waypoint mobility model at a speed of 5m/s. Figures 3.30, 3.31, 3.32 and 3.33 present the different metrics as a function of the network density through the network size.

When the network size increases, the network density i.e. the number of neighbors per node decreases. Therefore, it is expected that the tree or the mesh are less redundant. The figure 3.30 confirms this statement. Indeed, it shows that the PDR decreases slightly when the connectivity decreases. Moreover, as the density decreases, the network may experience some temporary disconnections.

Figure 3.31 illustrates the Control Bit Overhead with respect to the network size. It shows that for ODMRP, the Control Bit Overhead increases. Indeed, since each node as fewer neighbors, more copies of control packets are needed to reach all nodes in the network. For STAMP it remains stable and very low which seems to indicate that the structure presents stable redundancy.

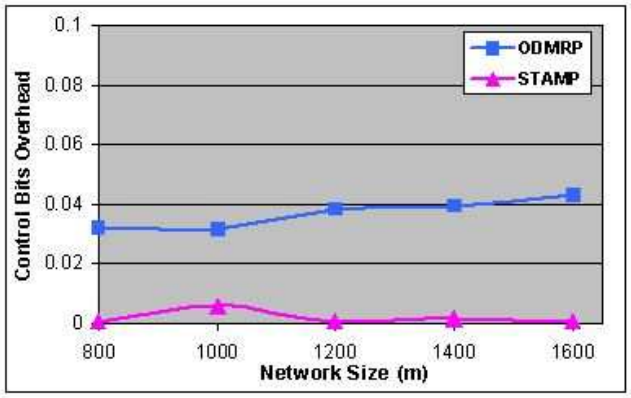

Figure 3.31 Control Bits Overhead Vs. Network Density

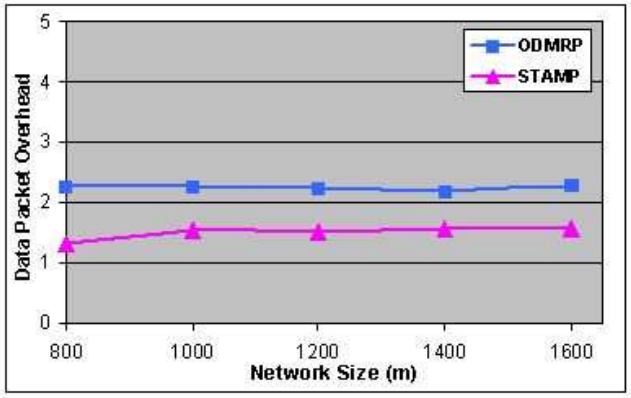

Figure 3.32 Data Packet Overhead Vs. Network Density

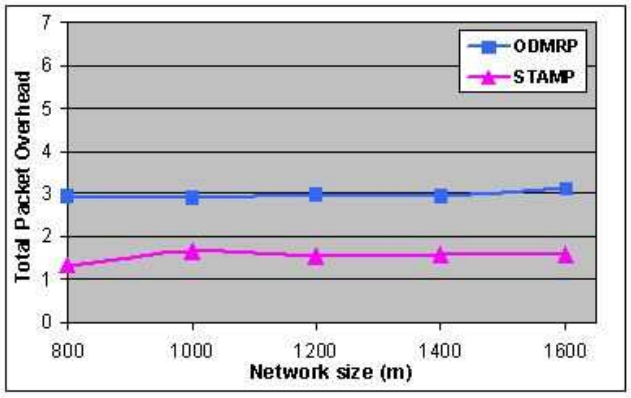

Figure 3.33 Total Packet Overhead Vs. Network Density

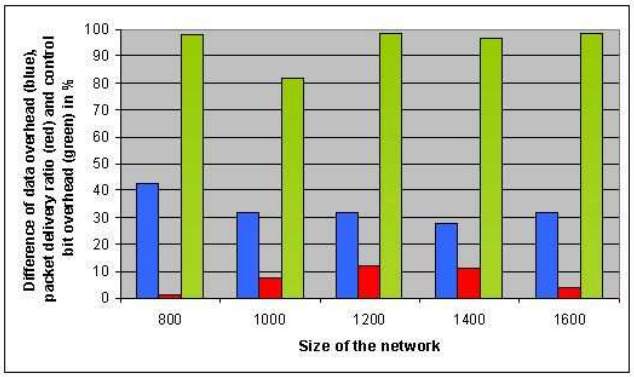

Figure 3.34 Comparison of the data overhead, the packet delivery ratio and the control bits overhead Vs. Network Density

3.5 Conclusion

In this chapter we proposed STAMP as a robust and efficient multicast routing protocol for intra-cluster multicast routing communications. We also present a new vision of the state-of-the-art of multicast routing protocol for MANET that takes the design objective as a criterion rather than a characteristic of the protocol such as the topology of the structure or the route acquisition scheme. This review underlines the lack of protocol that can provide high delivery guarantees with low overhead i.e. that is robust AND efficient. Consequently, the aim of STAMP is to propose an alternative to the existing flat multicast routing protocol by combining into one protocol the robustness and the efficiency compliance. The efficiency requirement is met thanks to a shared-tree structure maintained through a hard-state approach and where the initiative of the shared tree construction is given to the group members. Moreover, STAMP relies on an "hop-by-hop" branch construction to avoid multiple restarts of the joining process due to link breakages. To avoid the sending of redundant control information, STAMP re-uses as much as possible the information coming from the unicast routing protocol. In order to have a tree made of shortest paths from the members to the core only, each on-tree node periodically checks if its next hop node to the core is also its upstream node. During its construction or its maintenance process, STAMP does not employ any flooding or periodical message sending. For the robustness, STAMP takes advantages of the broadcast capacity of the medium to introduce redundancy without increasing the data overhead. STAMP benefits from the fact that two on-tree nodes may be neighbors and may belong to two different branches of the tree.

With the aim to justify our proposition and to verify that our objectives were reached, we fulfilled a performance evaluation in which a variety of mobility and network configurations are invoked. We employed different node mobilities, network sizes, multicast group configurations. To grade STAMP with respect to the other multicast routing protocols, we performed the performance comparison study by comparing the performance of STAMP with the performance of ODMRP. We chose similar scenarios of comparison than the one employed in the multicast comparison paper [78], so that we can compare the performance of STAMP with protocols evaluated in this paper.

The obtained results demonstrate outstanding features in favor of STAMP. In scenarios where the tree-based protocols are known to fail in term of packet delivery ratio, when the mobility increases for example, STAMP achieves high Packet Delivery Ratios comparable to the mesh-based ones. All the more that this high delivery guarantees are achieved with a high efficiency and not at the expense of a high data and control overhead as for mesh-based protocols.

For all the experiences we perform, STAMP presents similar Packet Delivery Ratios to ODMRP which is usually considered as the reference protocol for this metric. Moreover, the data packet overheads of STAMP are most of time inferior, around 30% or 40% less. In term of control overhead, STAMP presents in almost all scenarios a decline of more than 80%. These results prove that STAMP achieves its objectives in term of robustness and efficiency.

In this part, we presents STAMP as an alternative to the existing flat multicast routing protocols. Nevertheless, we remind that STAMP is also supposed to be employed as an intra-cluster multicast routing protocol in a clustered MANET. The following chapter presents the inter-cluster multicast routing protocol that should be implement to operate with STAMP in the tactical MANET.

Chapter 4

Inter-Cluster Multicast Routing Protocol

In chapter 2, we define the architecture of the multicast communications within the tactical network. We come to the conclusion that the clustering approach, which gathers nodes into groups, presents the most promising characteristics to achieve the scalability objective of the tactical MANET. The structure of a network in which clustering is applied has important repercussions on the architecture of the multicast service. Indeed, depending on the repartition of the actors of the multicast communications, the multicast service may be limited to a single cluster or may span over several clusters. Therefore, we distinguish two levels of multicast communications: the intra-cluster multicast communications when the multicast members are located within the same cluster and the inter-cluster multicast communications when the multicast members belong to different clusters. To handle such situations, a possible solution may be to define a global protocol that does not make any distinction between these two levels of multicast service. Nevertheless, such a solution would not benefit from the clustering structure. Our approach is to differentiate these two levels of multicast communications and to define a multicast routing protocol responsible for handling the multicast communications for each level. Therefore, it comes the need for an intra-cluster multicast routing protocol that is responsible for the multicast flows within each cluster and an inter-cluster multicast routing protocol that is responsible for the multicast flows from cluster to cluster. These two protocols interact to provide an end-to-end multicast service within the tactical MANET. The intra-cluster multicast routing issue has been addressed in the preceding chapter in which we propose and analyze STAMP, a robust and efficient shared-tree multicast routing protocol. In this chapter, we address the inter-cluster multicast routing issue. In a first part, the requirements on the inter-cluster multicast routing protocol are defined. Then, a review of the state-of-the-art in the field of multicasting with clusters is presented. In a third part, we describe the protocol we defined called ScAlable structure-Free Inter-cluster Multicast Routing SAFIR) protocol. Finally, the results of the discrete event performance evaluations we did on are proposed and analyzed.

We give here some useful definitions:

- A cluster: a subset of the network nodes designated by the clustering protocol that are represented to the other network nodes through a clusterhead.

- A clusterhead or a cluster leader: a special node within each cluster that is responsible to represent the nodes belonging to the same cluster than itself to the other network nodes.

- A clustering algorithm/protocol: the protocol that is responsible for gathering nodes into clusters and also to designate the clusterhead in each cluster.

- A clustered network: a network in which a clustering protocol is employed and consequently where network nodes are gathered into clusters.

4.1 Requirements on the inter-cluster multicast protocol

The requirements that are expected on the inter-cluster multicast routing protocol are similar to the one expected on the intra-cluster multicast routing protocol but have different impacts at the cluster level.

- Robustness: as long as the inter-cluster level is concerned, robustness i.e. the fact that the protocol operates correctly or delivers a good ratio of data packets to the destinations even in case of mobility, is an important requirement. Indeed, if a cluster does not receive data because of the protocol does not work properly, it means that no multicast members in this cluster will receive the multicast data. Since a cluster represents a set of nodes, a fault at the cluster level has an impact not on a single node but on a group of nodes and consequently, a single fault at the cluster level results in multiple faults at the node level. Nevertheless, if the robustness requirement is met thanks to redundant paths as in the "classical" flat intra-cluster multicast routing protocols, data overhead would be multiplied. Indeed, a virtual link between two clusters is made of several nodes, therefore, employing redundant paths between clusters means duplicating data on many nodes. We can thus say that the clustering solution has a "multiplicative" effect on the data overhead and on the fault impact.

- Efficiency and control overhead: Both the control overhead and the data overhead should be maintained as low as possible. Therefore, the number of control messages sent over the network or from cluster to cluster should be minimal. Moreover, as described previously, the data redundancy should be employed only when necessary.

- Energy consumption: the consumption of energy resources is directly linked to the amount of messages emitted and received. Therefore, optimizing the efficiency may be an interesting first approach to energy saving. Nevertheless, it is not enough. Indeed, balancing the control and data load over the nodes rather than concentrating all traffic on few nodes may also be considered unless some nodes have extra-resources.

- Scalability: The scalability requirement should be achieved thanks to the cluster structure. This assumption must be verified in this chapter through performance evaluation.

The clustering algorithm takes a part in the completeness of the preceding requirements and particularly on the robustness requirement. Indeed, if the clustering protocol does not operate correctly to update the cluster structure in case of mobility, the inter-cluster multicast routing protocol will not be able to react correctly to the mobility. Concerning the energy consumption, some clustering protocols propose to alternate periodically the cluster leader in order to balance the load of control messages. Moreover, in the tactical environment, nodes are supposed to move principally in groups. Therefore, the clusters are supposed to be relatively stable and

the evolution of the cluster topology is supposed to be slow. However, in each group, nodes are supposed to move. The robustness to mobility is therefore a requirement that aplies more on the intra-cluster routing protocol than on the inter-cluster routing protocol. Therefore, in this chapter, we will principally focus on the efficiency requirement.

4.2 Multicasting with clusters in MANET: state-of-the-art

The field of multicast routing protocols in MANET has been intensely studied over the last ten years. Most of the research works focus on flat networks made of around one hundred of nodes. Nevertheless, with the apparition of needs for MANETs made of several hundreds of nodes, the scalability has become an issue for all the firstly proposed protocols. Performance evaluations of these protocols have underlined limitations and arisen the need for new solutions of multicast routing in large MANET in order to address the scalability challenge faced by tactical MANET for example. We remind that we assume that the network is partitioned into clusters thanks to a clustering algorithm i.e. that the network is a "clustered network". Two classes of protocols can be distinguished among the protocols proposed in the literature to address scalability of multicast routing in MANET.

The first category is made of protocols that do not consider that the network is divided into clusters thanks to a clustering algorithm. These protocols (HMP [21], WCMRP [58], HDDM [52]) create a multicast delivery structure such as a tree and then divide it into sub-structures that are maintained locally. For example, the Weight-based Clustering Multicast Protocol (WCMRP) constructs firstly a source tree structure over the entire network thanks to a "Join_Request and Ack" process. Once the multicast tree is constructed, clusterheads are selected among the tree nodes. These clusterheads are responsible for the nodes under their sub tree in the multicast tree. The clusterheads are selected based on a weight criteria. Each sub tree is then maintained locally by each clusterhead. In the Hierarchical DDM (HDDM) protocol, the multicast group is partitioned into a given number of sub-groups of multicast members. Within each sub-group, a special node is chosen to serve as a sub-root. Then, the sub-group made of the source and all the sub-roots forms a special sub-group for the purpose of the upper level multicast. The DDM protocol is then applied in each sub-group at the initiative of the sub-roots and of the source for the upper level sub-group. The partitioning of the multicast group into multiple sub-groups is made thanks to the "classical" DDM protocol. It assumes that the source knows the exact list of all multicast members of the multicast group. The preceding protocols do not rely on a clustered network topology. We can imagine adaptations so that the partition of the multicast tree, that either WCMRP or HDDM are doing, matches the clustered topology of the network. Nevertheless, such protocols do not take benefits from the cluster topology. Moreover, these protocols do not make any distinction between the intra-cluster and inter-cluster multicast routing.

The second category consists of protocols that are designed for the purpose of inter-cluster multicast routing only. These protocols (MHMR [6], CBMRP [118],

MLANMAR [135], HIM-TORA [94]) are similar in the sense that they propose to apply flat multicast techniques such as the construction of a tree or a mesh on the cluster topology. The Mobility-based Hybrid Multicast Routing (MHMR) protocol proposes to construct a mesh structure defined as a subgraph of the clusterheads graph. Join_request messages are sent from clusterheads to clusterheads above the network. When a clusterhead that has members for the multicast group in its cluster receives a join_request message, it sends back a join_reply message. This join_reply message follows the clusterhead path back to the clusterhead that has sent the initial join_request message. The join_request and join_reply processes are repeated periodically to maintain the mesh structure. This procedure is similar to the one employed in flat networks to construct multicast mesh topologies. Similarly, the Cluster-Based Multi-Source Multicast Routing Protocol (CBMRP) proposes to construct a source-based tree of clusters. Nevertheless, applying the traditional flat approach on the cluster topology may lead to an important overhead. Indeed, as explained previously, the clustering has a "multiplicative effect". In this case, it means that sending a control message that just needs a single operation of sending in a flat network, implies in the cluster topology that the message is forwarded several times to go from one clusterhead to its neighbor clusterhead. Moreover, to reach all the neighbor clusterheads, a control message needs to be unicasted to each neighbor clusterhead whereas in a flat network, it only needs to be sent once to reach the neighbor nodes thanks to the broadcast capability of the medium. Therefore, a method such as constructing a tree that is efficient in flat networks may turn into a poorly efficient solution in clustered networks. Finally, such solutions are not efficient in term of control overhead because the multicast protocol is designed as a stand-alone protocol which means that it does not benefit from other services that may exist in the network such as clustering or unicast routing. Indeed, a multicast routing protocol is not a stand-alone service in a node architecture but is integrated in a network architecture where multiple services are deployed. All these services must be designed with the objectives to be optimized as a "whole". That way, the multicast routing protocol may take advantage of the control messages exchanged by other services to piggyback in these messages the information needed for its operation. The Multicast-LANMAR proposes such a solution. Based on the LANMAR routing protocol, it proposes to add the multicast group membership of each cluster in the landmark routing information exchanged by the unicast routing protocol. Therefore, each clusterhead knows the multicast membership of the other clusters. Then, the multicast data are unicasted from the source cluster to each of the clusters where there are members for the multicast group. The multicast data are thus duplicated as many time as the number of clusters where there are members. Unfortunately, even if the M-LANMAR proposes an interesting approach with the piggybacking of multicast information in the unicast messages, employing unicast tunnels presents a poor efficiency in term of data overhead. Moreover, the protocol is designed to operate only with the LANMAR protocol that assumes only group mobility which is too restrictive for our environment.

As a conclusion, it seems that the existing solutions of multicast routing designed to operate in a clustered network are not optimized. In the following part, we presents our solution that similarly to M-LANMAR benefits from the other services control messages to send the information needed for its operation. Therefore, our protocol

does not send messages to construct and maintain a multicast tree or mesh over the cluster topology. Moreover, the proposed protocol is independent from the unicast routing protocol and from the clustering protocol employed in the network.

4.3 Description of the ScAlable structure-Free Inter-cluster Multicast Routing protocol (SAFIR)

Our protocol called ScAlable structure-Free Inter-cluster Multicast Routing Protocol (SAFIR) defines a method for routing multicast information in a clustered, hierarchical network made of nodes that can be mobile. SAFIR is responsible for handling the inter-cluster multicast communications and assumes that an intra-cluster multicast routing protocol such as STAMP is applied within each cluster. Therefore, the aim of the protocol is to define how a multicast datagram for a group G can be forwarded from cluster to cluster until reaching the clusters where the multicast members for the multicast group G are. This objective brings to the fore several questions that must be considered by our protocol:

- How can a node know the list of clusters where the multicast members for a group G are?
- How can a clusterhead know which neighbor clusterheads a multicast datagram needs to be forwarded to?
- Which node is responsible for the forwarding decision in a cluster?
- Which information is the decision to forward a multicast datagram based on?
- How is the interconnection between the intra and the inter cluster multicast routing protocols made? Which information must be exchanged ? Which node is responsible for this function ?

SAFIR has been designed with the objective to be independent from the intra-cluster multicast routing protocol employed. Therefore, any type of flat multicast routing protocol can be used, even the broadcast solution is possible. Nevertheless, we design STAMP with the aim to employ it in conjunction with SAFIR. The association SAFIR/STAMP is more than possible, it is recommended. Nevertheless, in this chapter, SAFIR is presented without any dependence on STAMP.

In SAFIR, as in almost all inter-cluster multicast routing protocols, the clusterhead has the responsibility for deciding whether a multicast data packet must be forwarded or not. Moreover, SAFIR is characterized in that it does not need any join or leave messages to be exchanged between the clusterheads or gateway nodes to create a structure above the clusterheads as it is the case in MHMR or CBMRP. Indeed, rather than constructing a structure allowing to distribute the multicast data among the clusters, SAFIR defines a method where each clusterhead takes its own decision to forward or not data packets autonomously so that each data packet makes its own path when it goes from cluster to cluster. Such a solution is based on the fact that in each cluster, the clusterhead knows two pieces of information:

- the multicast group membership of the nodes belonging to its cluster. A clusterhead does not know which are the addresses of the nodes that are members for a multicast group in each cluster. It only knows that members for a specific multicast group are present and that consequently multicast data needs to be forwarded to this cluster. It is then up to the intra-cluster multicast routing protocol to distribute the multicast datagrams to the members.
- the addresses of its neighbor clusterheads.

Unicast routing protocols developed for clustered networks as well as clustering protocols present a common characteristic which is that the clusterheads periodically share routing-related information among themselves. Therefore, our protocol follows the same approach so that the messages of both unicast and multicast routing protocols could potentially be aggregated. The two pieces of information mentioned above are piggybacked in unicast or clustering control messages exchanged between the clusterheads so that:

- Each clusterhead is aware of the multicast cluster membership of all clusters in the network.
- Each clusterhead constructs a distance vector cluster routing table (first solution) or cluster link state database (second solution). We present both solutions.

With these two pieces of information only, when a clusterhead receives multicast data, it is able to decide on its own if it has to forward data to other clusterheads and if so, to which neighboring clusterheads.

In the remainder of this part, we describe the operation of SAFIR firstly when a distance vector approach is taken for the cluster routing table and secondly when a link state approach is preferred.

4.3.1 First solution: Distance Vector approach

In this first solution, each clusterhead constructs a distance vector cluster routing table of the cluster topology.

4.3.1.1 Construction of the distance vector cluster routing table

In this part, we explain how the distance vector cluster routing table is constructed.

Thanks to the clustering protocol, each clusterhead knows the nodes belonging to the cluster it represents. It is also aware of the identity of the neighboring clusterheads. More specifically, two clusters are regarded as neighbors if they contain at least a couple of neighboring nodes (one in each cluster). Therefore, similarly to a distance vector routing protocol like DSDV, each clusterhead is going to exchange periodically with its neighboring clusterheads its distance vector cluster routing table.

This table contains one entry per known destination clusterhead. Each entry is composed of three values i.e. the id of the destination clusterhead, the id of the clusterhead of the next hop cluster to reach the destination clusterhead and the number of cluster to go through to reach the destination clusterhead. This

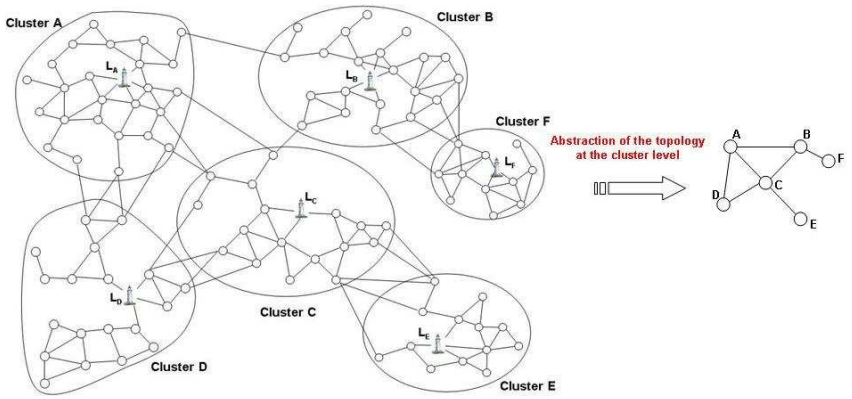

Figure 4.1 Cluster Topology Abstraction

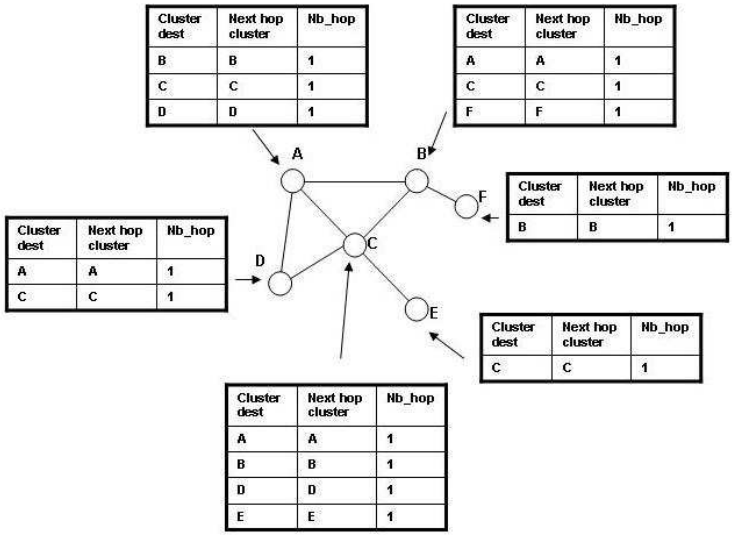

Figure 4.2 Construction of the Distance Vector Table Step 1

configuration is minimal. Additional fields such as other metrics can be added for QoS purposes for example.

Let us consider the network of figure 4.1, where a clustering algorithm has been applied. If we abstract the topology at the cluster level, cluster A has 3 neighbor clusters D, C and B, cluster D has two neighbor clusters A and C, and so on. The obtained abstraction of the topology is illustrated by the right part of figure 4.1.

Considering this topology, the process to construct the distance vector table will be the following:

- First step (figure 4.2): each clusterhead sends its table to its neighbor clusterheads. At this step, each clusterhead only knows its neighbor clusters.
- Second step (figure 4.3): upon reception of these messages, each clusterhead updates its distance vector cluster routing table and then sends it to its neighbor clusterheads.
- Third step (figure 4.4): upon reception, each node updates its table. Since

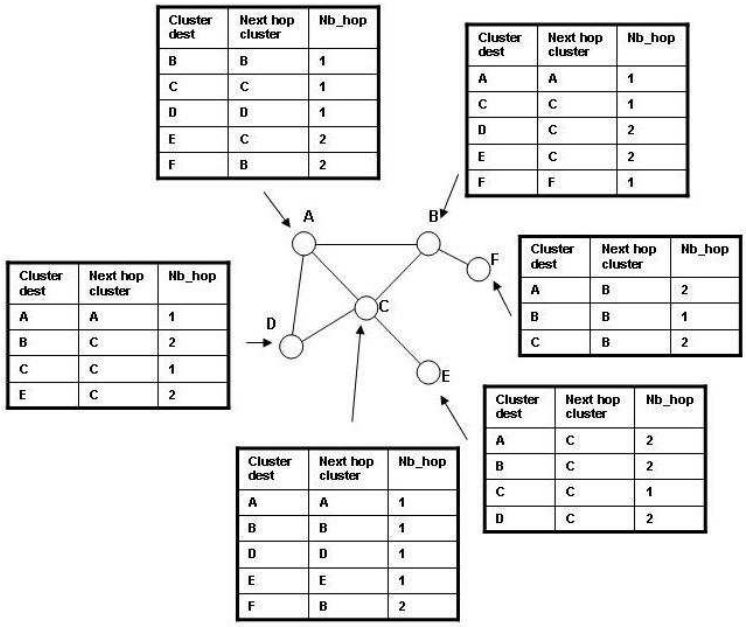

Figure 4.3 Construction of the Distance Vector Table Step 2

node E, F and D receive new information, they forward their new table.

Each clusterhead has now a distance vector view of the "cluster network". Each time a clusterhead detects changes in its cluster neighborhood, it sends an update message so that the global topology may be known by every other clusterheads. It is a triggered event mechanism.

4.3.1.2 Group membership exchange

This second part of the algorithm assumes that each clusterhead knows the multicast membership of the nodes belonging to its cluster i.e. it must know which are the multicast groups for which there are members in its cluster (Assumption 1). The exact list of each node membership is not needed.

Let us consider the example of figure 4.5. Node L_D (clusterhead of cluster D) must know that in its cluster, there are multicast members for the multicast groups 1 and 2. Node L_E must know that there is no node in its cluster belonging to a multicast group. Node L_A must know that there are multicast members for the multicast group 1 in its cluster.

As well as the distance vector cluster routing table, the clusterheads exchange their cluster multicast membership table so that each clusterhead knows the multicast membership of all other clusters. Periodically, the clusterheads exchange the delta between their current table and their last exchange. This table contains one entry per known multicast group and each entry associates a multicast group address to the id of the clusterhead of the clusters where there are members for this multicast group.

Even if SAFIR is not responsible for defining how the assumption 1 can be achieved, some propositions can be made:

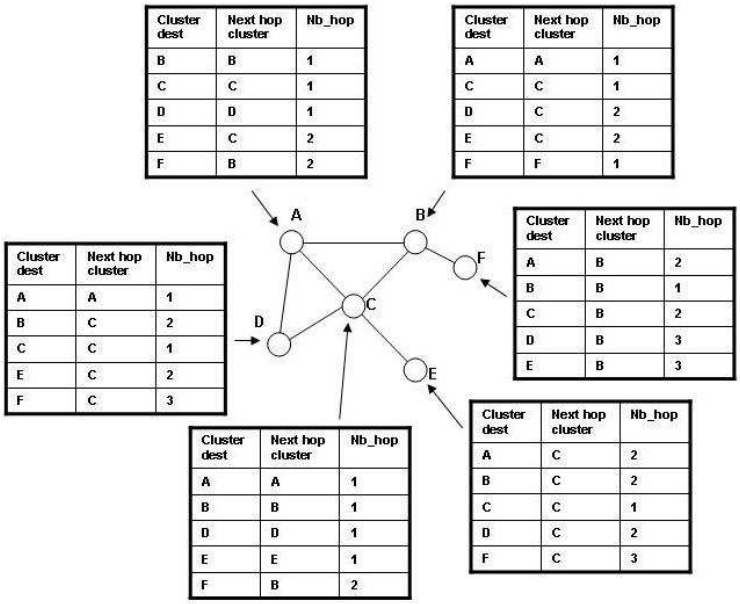

Figure 4.4 Construction of the Distance Vector Table Step 3

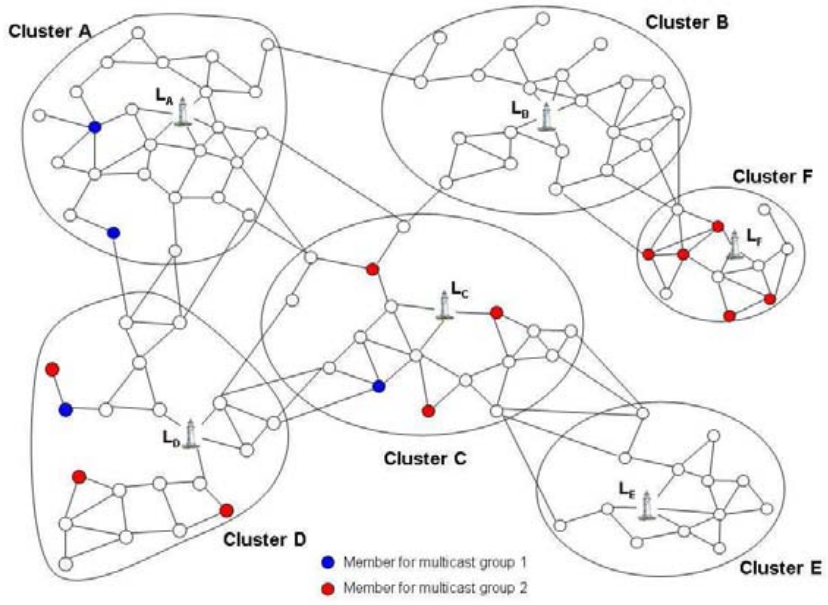

Figure 4.5 Example of a multicast member repartition

- If an intra-cluster multicast routing protocol is used, and if it is a shared-tree protocol, the clusterhead can be chosen as the core node. Each time the clusterhead receives a join message; it knows that there is at least one node of its cluster that is member of the related multicast group. This solution is the best one in term of data and control overhead, all the more if STAMP is chosen as the intra-cluster multicast routing protocol.

- It is common in military networks that the mission is preplanned and therefore that every node is aware of the multicast membership of all other nodes in the network. Therefore, since a clusterhead knows the list of the nodes that belong to its cluster, it may know the multicast membership of its cluster.

- Another solution can be that each node periodically announces its multicast membership to its clusterhead. This solution may generate an important additional overhead and is therefore not recommended.

4.3.1.3 Data forwarding

This part refers to the processing that should be done when a multicast data packet is received. Depending on the status of the node (source node, ordinary cluster node[1], or clusterhead), the treatment can be different.

When a source has multicast data to transmit, it just sends the multicast datagrams to its clusterhead. Upon reception of the datagrams, the clusterhead must determine which neighbor clusterheads it has to forward the multicast datagrams to. To take this decision, the clusterhead searches through its cluster multicast membership table which clusters are "cluster members" for the multicast group identified as the destination address of the multicast datagram. For each cluster of the list, the clusterhead determines thanks to its distance vector cluster routing the Next Hop clusterhead to reach the destination cluster. At the end of this operation, the clusterhead knows the Next Hop clusters to which it has to forward the multicast datagrams.

To sum up, when a clusterhead receives a multicast data packet:

- If packet already received
  - Discard it and stop the forwarding process for this packet
- Else
  - Search for the Next Hop clusterhead as described previously.
  - If one on the Next Hop clusterheads is the one through which the data has been received
    - Remove this cluster from the list of the clusters to forward the datagram to
  - Forward the packet to the clusterheads of the list.

Let us consider the example of figure 4.6. Node S_6 has multicast datagram to send for multicast group "green". S_6 sends the data packet to its clusterhead L_E

[1] an ordinary cluster node is a node of a cluster that is neither a source nor a clusterhead

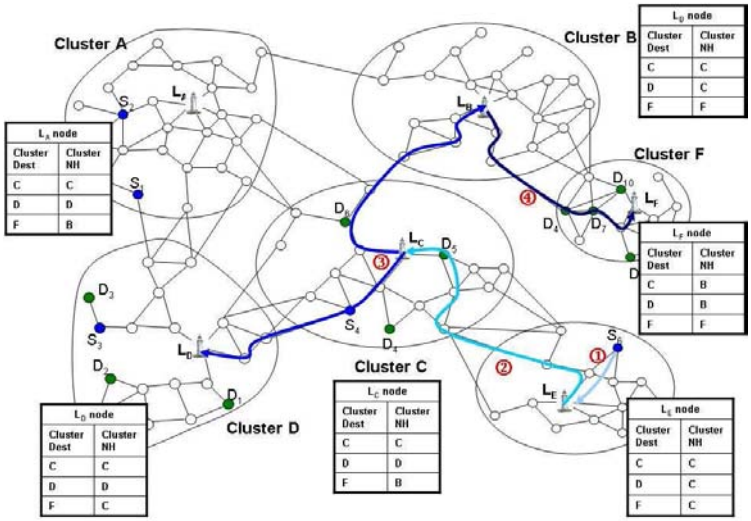

Figure 4.6 Example of data forwarding in the distance vector solution

(1). Upon reception of the multicast data packet, L_E looks in its cluster multicast membership table. Clusters C, D and F are "cluster members". From its distance vector cluster routing table, L_E knows that the Next Hop Cluster for each of these three clusters is cluster C. Therefore, L_E forwards the multicast packets to L_C, the clusterhead of cluster C (2). Upon reception, L_C looks at its cluster multicast membership table. First of all, since cluster C is itself a cluster member, L_C "gives" the multicast datagram to the intra-cluster multicast routing protocol for forwarding within the cluster. Then, clusters D and F are "cluster members". The Next Hop cluster for cluster D is cluster D and the Next Hop cluster for cluster F is cluster B based on the information contained in the distance vector cluster routing table. Therefore, L_C forwards the data packets to both L_B and L_D (3). Upon reception of the data packet, L_B looks at its cluster multicast membership table. Clusters C, D and F are "cluster members". From its distance vector cluster routing table, L_B knows that the Next Hop Cluster for cluster F is cluster F. Therefore, it forwards the data to L_F. For clusters C and D, the distance vector cluster routing table says that the Next Hop cluster is cluster C. Since data packets have been forwarded to L_B by cluster C, L_B does not forward the data packet to L_C. Upon reception of the data packets, both L_D and L_F forward it on their cluster. Since the Next Hop cluster to reach the other cluster members is the one that has forwarded the data to them, i.e. L_C and L_B, L_D and L_F do not forward the data to another cluster. The process is over.

The path followed by a data packet is independent from the one followed by other packets. Moreover, the end-to-end path followed by each multicast datagram draws a sort of mesh, whose redundancy depends on the cluster connectivity.

Therefore, the global "structure" followed by the multicast datagrams can be common or specific to all the multicast groups and to all the multicast sources of a multicast group depending on the multicast members repartition over the network.

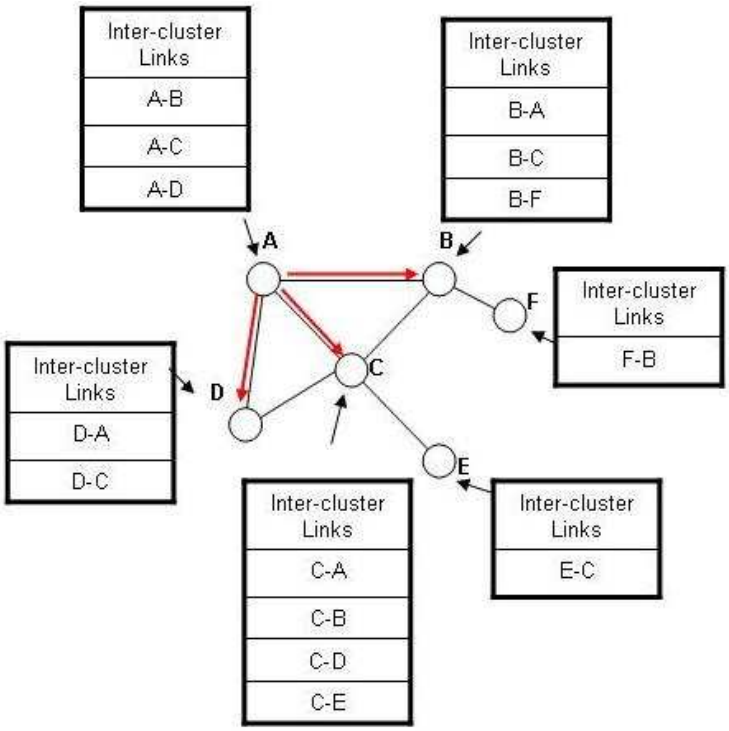

Figure 4.7 Construction of the inter-cluster link state database step 1

4.3.2 Second solution: Link State approach

In this second solution, each clusterhead constructs a link state database of the inter-cluster links of the topology. An inter-cluster link is defined such that if two clusters are neighbors, it exists an inter-cluster link between these two clusters and the inter-cluster link is identified by the address of the clusterheads of the two neighbor clusters.

4.3.2.1 Construction of the link state database

In this part, we explain how the database of the inter-cluster links of the topology, refer as the link state database is constructed by each clusterhead.

Thanks to the clustering protocol, each clusterhead may know which clusters are its neighbor clusters. Therefore, similarly to a link state routing protocol, each clusterhead sends to all other clusterheads in the network its inter-cluster link states. Each time a clusterhead receives an inter-cluster link state message, it updates its link state database. This base contains one entry per link. Each entry is composed of the addresses of the two clusterheads that make the inter-cluster link.

The process to construct the link state database is the following one.

- Firstly, each node updates its database to register its links with its neighbors.
- Then each node broadcasts to its neighbors its inter-cluster link state message.

Let us consider the network illustrated by figure 4.1 as in the distance vector solution. Note that we only focus on the inter-cluster link state message from cluster

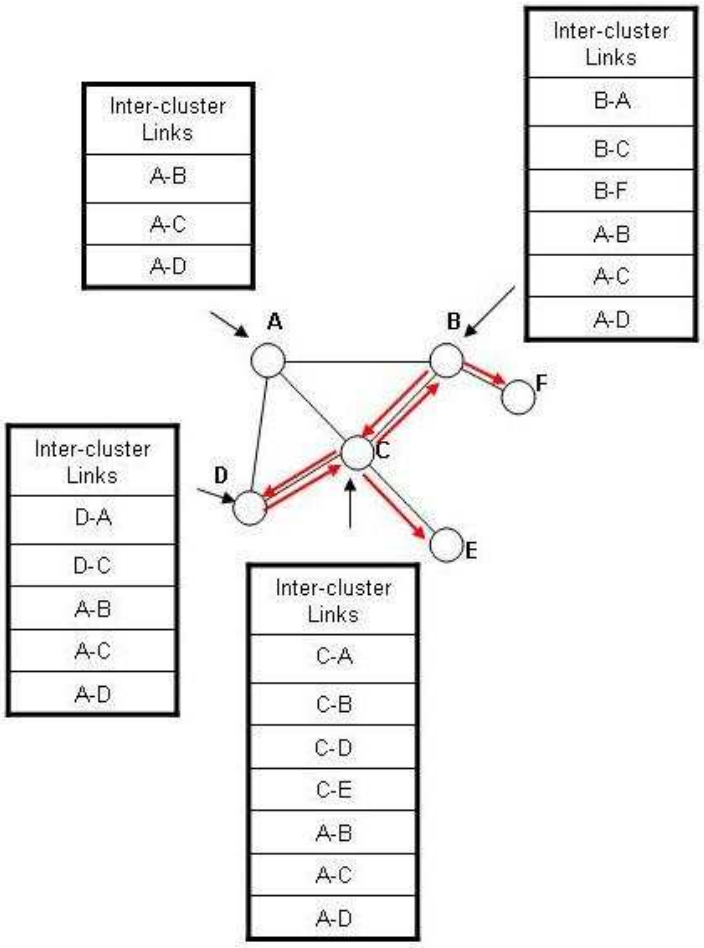

Figure 4.8 Construction of the inter-cluster link state database step 2

A (red arrows). As illustrated by figure 4.7 where the network is represented only by the abstracted cluster topology, A sends the message to its neighbors B, C and D. Upon reception, B, C and D update their database, and forward the message to their cluster neighbors (figure 4.8). Finally, when all messages have been forwarded in the network, all clusterheads shared the same inter-cluster link state database (figure 4.9).

Remark: in order to save bandwidth the message is not forwarded to a clusterhead from which it has been received (here cluster A). Similarly, a clusterhead does not forward a message that it has already forwarded before.

4.3.2.2 Group membership exchange

It is the same process than the one described before for the distance vector approach.

4.3.2.3 Data forwarding

This part refers to the treatment that should be done when a multicast data packet is received. Depending on the status of the node (source node, ordinary cluster node

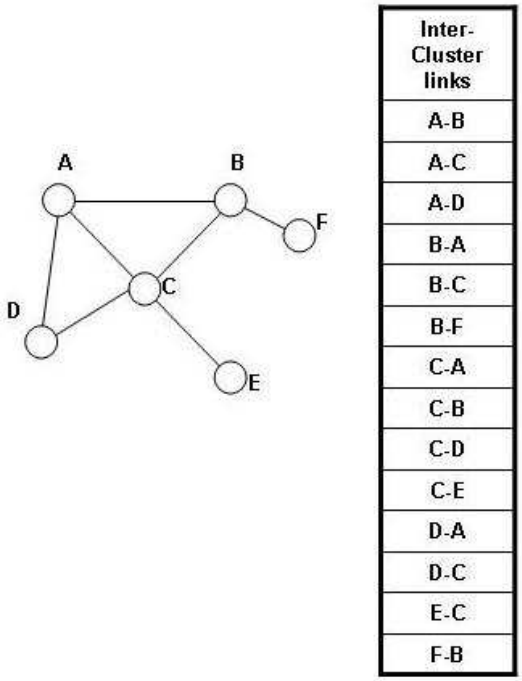

Figure 4.9 Construction of the inter-cluster link state database step 3

or clusterhead), the treatment can be different.

When a source has multicast data to transmit, it sends the multicast datagrams to its clusterhead. Upon reception of the datagrams, the clusterhead must find which neighbor clusters it has to forward the data to. To make this decision, the clusterhead searches through its cluster multicast membership table which clusters are the "cluster members" for the multicast group identified by the destination address of the data packet. To make this decision, the clusterhead determines the cluster path between itself and the clusters members. It uses the inter-cluster link state database to determine this path. Therefore, the clusterhead is able to determine a sort of multicast source tree according to the algorithm presented in figure 4.10. It forwards the multicast packets to each cluster neighbor on the different "tree branches". When receiving a multicast packet from another clusterhead, a clusterhead computes the source tree rooted on the cluster the source of the message belongs to. The clusterhead is thus able to determine which branches of this tree it belongs to. Therefore, it forwards the data to its next hop clusterhead on these branches. This information, i.e. which neighbor clusterhead a clusterhead may forward to, is kept in cache so that the clusterhead may not re-do the calculation each time a packet is received.

Let us consider the example of figure 4.11 where node S_2 has multicast packets to send for the multicast group "green". Clusters C, D and F have multicast members for this group. Therefore, in their multicast membership tables, the clusterheads have clusters C, D and F identified as cluster members for group "green". S_2 sends its multicast data packet to its clusterhead L_A (1). Upon reception of the packets, L_A computes the cluster path from cluster A to clusters D, C and F. The paths are the following ones: $A \rightarrow D$, $A \rightarrow C$, $A \rightarrow B \rightarrow F$. Therefore, L_A forwards the

```
C = list of clusters of the network not included in any path
C_m = list of clusters where there are members and for
      which a path still need to be found
L = set of "cluster links"
S = cluster source
paths = set of the computed paths
PathsList = list of paths from the source cluster S to the cluster members
pathsNew = the new list of path computed in a round

Initialisation:
Paths = {[S]}
C = C-{S}
C_m = C_m-{S}

While C_m ≠ Ø
   For all path P of Paths
      Neighbor = list of neighbors of the last cluster of P
                 (This list is computed thanks to L)
        For all cluster C_n of Neighbor
           If C_n ∈ C
              Create a new_path = path + { C_n } *
             pathsNew = {pathsNew, { new_path }}
              C = C-{ C_n }
              If C_n ∈ C_m
                 PathsList = {PathsList, {new_path}}
                 C_m = C_m - { C_n }
              End If
           EndIf
        EndFor
   EndFor
   Paths = pathsNew
EndWhile

* "+" refers to the concatenation operation
```

Figure 4.10 Algorithm to determine the Next Hop clusters in the link state solution

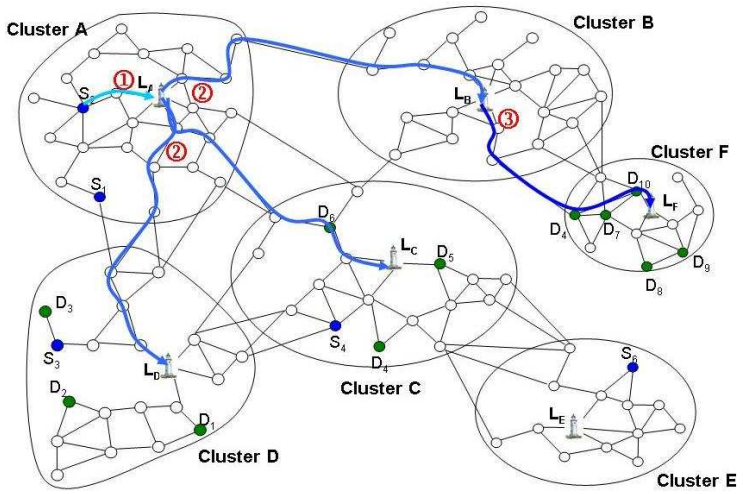

Figure 4.11 Example of data forwarding in the link state solution

multicast packet to clusterhead L_C, L_D and L_B (2). Upon reception of the multicast data, L_C, L_D and L_B identify the "cluster source" of the data, L_A. L_C, L_D and L_B compute the paths between cluster A and clusters C, D and F. C is only a last hop in one, therefore it does not forward the data to another clusterhead. D is also only a last hop in one path; therefore it does not forward the data to another clusterhead. B belongs to the path between A and F, therefore, it forwards the multicast data to the next hop cluster on this path, L_F (3). Upon reception of the multicast data, L_F identifies the "cluster source" of the data, L_A. L_F computes the paths between cluster A and clusters C, D and F. F is only a last hop in one path, therefore it does not forward the data to another clusterhead.

4.3.3 Comparison of the Link State and the Distance Vector solutions

If we compare the link state and the distance vector solutions, we can remark a difference in the redundancy of the paths followed by the datagrams from cluster to cluster. Indeed, with the link state approach, the clusterheads have a global view of the cluster topology which enables each clusterhead to compute a source tree from the cluster source to all cluster members. With the distance vector solution, each clusterhead has only a local view of the cluster topology. Therefore, a clusterhead cannot know to which other clusterheads the packets have already been forwarded. Considering the example on figure 4.12, the data forwarding with the distance vector approach would have introduced some redundancy (red arrows) in the paths followed by the data packets. When L_C receives the multicast packets from L_A, L_C does not know that L_A has already forward them to L_D. Thus, L_C forwards the multicast datagrams to L_D since D is the next hop cluster to reach the cluster D which is a cluster member. It appears here that there is a trade-off between the efficiency and the robustness. If efficiency is preferred, the link state solution should be chosen whereas if robustness is valued, the distance vector approach should be chosen. As previously explained, we consider that the inter-cluster multicast routing protocol

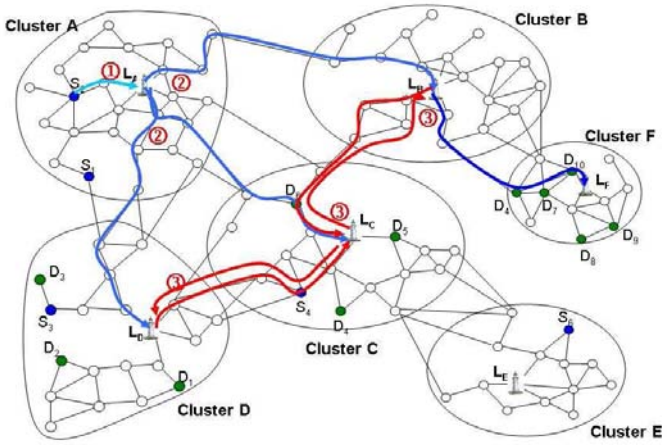

Figure 4.12 Comparison between Link State and Distance Vector approach

is more concerned by efficiency than by robustness. Therefore, in the performance evaluation part, we will only evaluate the link state solution.

4.3.4 Interconnection between the intra and the inter cluster multicast routing protocol

In the architecture of the multicast service that we define for the tactical MANET, we distinguish two levels of communications, the intra and the inter cluster multicast communications. These two levels are handled by two multicast routing protocols, an intra and an inter cluster multicast routing protocol. To provide an end-to-end multicast service within the tactical MANET, these two protocols must interact to exchange the information needed and the responsibility for forwarding the multicast datagrams. The interconnection of these two protocols is done at the clusterhead level in each cluster. It is a bidirectional communication, i.e. control information and/or data packets are passed from the intra-cluster to the inter-cluster multicast routing protocol or vice versa. Previously in this chapter, we present SAFIR as our solution for the inter-cluster multicast routing protocol. In this part, since SAFIR can be employed with any intra-cluster multicast routing protocol, we describe firstly the general case of the interconnection of SAFIR with any intra-cluster multicast routing protocol. Then, the interconnection of SAFIR and STAMP as the intra-cluster multicast routing protocol is presented. Indeed, one of the aim of STAMP is to be employed as an intra-cluster multicast routing protocol. The association SAFIR/STAMP is thus optimized.

4.3.4.1 General case

In this general case, any flat multicast routing protocol can be considered for interconnection with SAFIR, a mesh-based protocol, a shared-tree protocol, a source-tree

protocol, broadcast, etc. Two types of data can be exchanged between the two levels of multicast protocols.

The first one is related to the multicast membership of the nodes belonging to the cluster. As said previously in the description of SAFIR, we assume that the clusterhead is aware of the multicast membership of its cluster. Depending on the type of protocol employed, this information can be provided by the intra-cluster routing protocol. This can be done only if the protocol is a receiver initiated protocol. Indeed, if a shared-tree multicast routing protocol or a mesh-based protocol relying on a core node such as CAMP is employed, the clusterhead can be chosen as the core node or rendezvous point by default for all multicast groups. That way, each time the clusterhead receives a join message for a new multicast group G or a leave message for a multicast group G, the intra-cluster routing protocol can inform the inter-cluster routing protocol that respectively there are nodes in the cluster that are members of G or that there is no member anymore of G. For instance, an alert mechanism can be employed for this purpose. In the case of a source-initiated protocol, the responsibility to construct the structure is given to the source node which generally floods a message into the cluster to initiate the structure. Consequently, the clusterhead is only able to learn the presence of sources and not the presence of members. Therefore, with other types of multicast protocols (other than receiver-initiated), an additional mechanism needs to be defined so that the clusterhead may be aware of its cluster multicast membership.

The second type of data exchanged by the two levels of multicast protocol is the multicast datagram themselves. Considering the role of the clusterhead, two situations must be distinguished. The first case to consider is when the source belongs to the same cluster. When there is a source for a multicast group in a cluster, even if the intra-cluster multicast routing protocol is a source-initiated protocol, it is not the intra-cluster multicast routing protocol of the source node that handles the multicast datagram received from the multicast application but rather the inter-cluster multicast routing protocol. This means that the multicast datagram should be sent from the source node to the clusterhead without any care to the intra-cluster multicast routing protocol. Then, when the multicast datagram reaches the clusterhead, the inter-cluster multicast routing protocol looks in its cluster multicast membership table to identify the cluster members. If the cluster is identified as a cluster member, then the clusterhead passes a copy of the multicast datagram to the intra-cluster multicast routing protocol for forwarding within the cluster. If the intra-cluster multicast routing protocol is a source-initiated protocol, then the clusterhead initiates the construction of the multicast structure. Consequently, the clusterhead is seen by each cluster node as the source node of all multicast groups. If the protocol is a group-initiated protocol, it means that a structure is already constructed within the cluster. The multicast datagram has just to be forwarded on this structure. The second case is when the source belongs to another cluster than the clusterhead. The datagrams are thus received from another clusterhead, following SAFIR operation. When a clusterhead receives a multicast datagram from another clusterhead, it looks in its cluster multicast membership table to identify the cluster members. If the cluster is identified as a cluster member, then the clusterhead passes a copy of the multicast datagram to the intra-cluster multicast routing protocol for forwarding within the cluster.

4.3.4.2 STAMP as the intra-cluster multicast routing protocol

In this part, we describe the interconnection between STAMP and SAFIR as well as the adaptations that must be performed to STAMP to employed it in conjunction with SAFIR (and not as a stand-alone or flat protocol).

When STAMP is employed within a cluster in conjunction with SAFIR, the clusterhead acts as the core node for all multicast groups. Therefore, the core announcement process is no more needed and it is not the first node becoming member of a multicast group that becomes the core for the group. That way, when the clusterhead receives a join message for a multicast group G meaning that there are member nodes for this group in the cluster, STAMP passes this multicast membership information to SAFIR. SAFIR is thus aware of the multicast membership of nodes belonging to its clusters. Similarly, when the clusterhead receives a leave message for a multicast group G and when its downstream list associated to G is empty, the STAMP process must inform the SAFIR process that there is no member any more for G so that G can be removed from the cluster membership table.

When a source node has multicast data to send to a group G, the multicast packet must be processed by STAMP and not by SAFIR as in the general process. This optimization can be done because we are certain that whether there are members in the cluster or not, the multicast packets will finally reach the clusterhead thanks to the operation of STAMP. Indeed, whether there are members or not, with STAMP, when a source has multicast data to send to a multicast group, it forwards the data to its next hop on the path to the core (here the clusterhead) until reaching the clusterhead or an on-tree node. Consequently, if there are members in the clusters for this multicast group, and if it exist on-tree nodes on the path from the source to the clusterhead, then the data packet will be forwarded on the tree, and will finally reach the clusterhead since the clusterhead belongs to all multicast trees. In all other cases, the multicast packets will be forwarded hop-by-hop by all nodes on the path from the core to the clusterhead.

When the STAMP process in a clusterhead node receives a multicast packet, the STAMP process must pass to data packet to the SAFIR process of the node. Upon reception of a multicast datagram from the STAMP process, the SAFIR process performs a look at its cluster membership table to find the cluster members for the multicast group. Even if the cluster is identified as a cluster member, the packet must not be passed back to the STAMP process since it has already processed it. Nevertheless, even if this last check is not performed by the SAFIR process, meaning that the SAFIR process passes back the data packets to the STAMP process, STAMP will not forward the data packet since a verification is performed on the redundancy of the received multicast packets.

When a clusterhead receives a multicast packet from another clusterhead, the SAFIR process looks in its cluster multicast membership table for the cluster members. If the cluster is identified as a cluster member, then, the SAFIR process must pass a copy of the multicast packets to the STAMP process. Upon reception, the STAMP process forwards the data packets on the tree which must already be constructed. Indeed, the clusterhead has identified its cluster as a cluster member which means that the clusterhead is aware that there are multicast members for this multicast group, which means again that the clusterhead has received at least one join message for this group and thus that the tree is constructed.

In the beginning of this chapter, we present SAFIR, an inter-cluster multicast routing protocol. We also explain how SAFIR can interact with any intra-cluster multicast routing protocol. In the remainder of the chapter, we present the performance evaluation we realize on SAFIR. In this study, we implement the link state version of SAFIR.

4.4 Performance evaluation of SAFIR

In the previous parts of this chapter, we present our proposal, the ScAlable structure-Free Inter-cluster multicast Routing protocol, as a solution for routing the multicast flows from cluster to cluster in a tactical MANET where a clustering protocol is employed in order to meet the scalability requirement. In this part, we evaluate the performance of SAFIR through discrete event simulations. The goal of this evaluation is firstly to confirm the already-known advantages related to the use of a hierarchical clustered network with respect to a flat network, when the number of nodes in the network increases. Then, even if the robustness is not a principal requirement on the inter-cluster multicast routing protocol, the performance of SAFIR with respect to mobility is evaluated. Finally, we evaluate the performance of SAFIR associated with STAMP as the intra-cluster routing protocol and we compare it with, on the one hand, the association SAFIR and ODMRP, and on the other hand, ODMRP only. The rest of this section is organized as follows. In a first part we present the framework of this performance evaluation. In the second part, the metrics that are observed to analyze and to compare the protocols are given. Finally, the scenarios we work on for each performance goal are described and the results are commented.

4.4.1 Framework

We perform discrete event simulations thanks to the OPNET Modeler 11.5 simulator [95]. We model the Max-Min D-Cluster [5] protocol for the clustering. This protocol creates stable clusters since the clusterheads election is based on the node ID which is a non-evolving parameter rather than on a variable parameter such as the connectivity for instance. The control overhead of SAFIR includes the clustering control overhead of the Max-Min D-Cluster protocol in all the following simulations. For the comparison of SAFIR with ODMRP, we rely on the model of the ODMRP protocol that has been provided by a third party [2] and that has been slightly modified to be compliant with the IETF Internet Draft [136]. In the simulation where STAMP or ODMRP are employed as intra-cluster multicast routing protocol in conjunction with SAFIR, we slightly modify the models of the protocols to make them compliant with the design of SAFIR following the recommendations given in paragraph 4.3.

Our network is composed of mobile nodes with a radio range propagation of 250m randomly placed within a flat area. The number of nodes and the size of the network is dependent on the scenario. However, for all scenario we work on, the density remains constant. It is fixed to 50 nodes per $1000m^2$. The OLSR protocol is employed as the underlying unicast routing protocol when STAMP is employed in conjunction

[2] We thank MAJ Fernando J. Maymi, Assistant Professor in the Dept. of Electrical Eng. & Computer Science of the U.S. Military Academy, West Point for providing his model of ODMRP.

Table 4.1 OLSR simulation parameters values

Parameter	Value
Hello Message Interval	2 seconds
Topology Control Message Interval	4 seconds
Neighbor Hold Time	6 seconds
Topology Hold Time	12 seconds

with SAFIR with the parameters defined in the table 4.1. No unicast routing protocol is employed in the simulation with ODMRP. For the MAC layer, the 802.11 WLAN using Distributed Coordination Function is used with a channel capacity of 2Mbits/s. The Direct Sequence Spread Spectrum is employed as the modulation technique. The buffer size of the MAC layer is of 256 Kbits. Each node moves randomly according to the Random Waypoint model with no pause time [16, 86]. At the beginning of the simulation, each node selects a random destination in the area and moves to this destination at a speed defined by a parameter of the mobility model. Upon reaching this destination, it randomly chooses another destination and moves to this destination without waiting. The speed can be chosen randomly between two boundaries for each segment or may be fixed at the beginning of the simulation. The default speed value is set to 2m/s for all scenarios except those on which the robustness to mobility is evaluated. One can argue that this speed is too low. Nevertheless, in a tactical MANET, nodes are supposed to have a propagation range between 5 km to 10 km whereas it is set to 250 m in our experiments. Thus, a speed of 2 m/s in our experiments corresponds roughly to a speed of 144 km/h with a tactical node that has a propagation range of 5 km, which is quite an important speed. The multicast traffic is a Constant Bit Rate traffic where the size of a packet is 512 bytes. For each scenario, multiple runs of 500s with different seeds were run. The results are then averaged on these different runs. The time needed to execute one simulation run grows with the size of the network. For example, it takes several hours to execute a simulation with 500 nodes. Therefore, the number of runs that we can perform for each simulation scenario is limited. This limitation inherent to discrete event simulators with large networks explains the modifications of concavity that can be observed in some of the presented results.

4.4.2 Metrics observed

We follow the suggestions of the IETF MANET working group [30] for evaluating routing or multicasting protocol. The following metrics are chosen:
The Packet Delivery Ratio: the number of the received packets divided by the number of packets expected to be received, where the number of packets expected to be received is the number of data packets sent by the sources times the number of receivers.
The Data Packet Overhead i.e. the number of data packets transmitted on the network per data packet delivered, it measures the number of individual copy of data packets transmitted on the whole network.
The Control Bit Overhead i.e. the number of control bits transmitted per data bits delivered, it measures the control overhead needed to install and maintain the tree structure with respect to the data delivered.
The Total Packet Overhead i.e. the total number of packets (control and data)

transmitted on the network per data packet delivered, it measures the total number of packets to be transmitted on the network (control and data) to achieve the transmission of the datagrams.

We remind that we are mainly interested in the efficiency of the SAFIR protocol rather than in its robustness. Therefore, among the preceding metrics, we will provide higher attention to the Control Bit Overhead and the Data Packet Overhead.

4.4.3 About the interest of clustering

In this first experiment, we demonstrate the scalability of our protocol with respect to the number of nodes. We compare the use of the ODMRP protocol on a flat network with the use SAFIR with ODMRP as the intra cluster multicast routing protocol on a clustered network. In these scenarios (cf. table 4.2), there is one multicast group, 5 sources and 40 percent of the nodes are members of the group. Each node moves following the Random Waypoint mobility model with no pause time at a speed of 2m/s.

As shown in figure 4.13, the PDRs are similar in both protocol configurations. One can expect that the PDR of ODMRP falls when the number of nodes increases. Such a decline is traditionally observed when the load of the network increases with the number of nodes. Since we want to see the influence of the protocol operation without being influenced by the capacity of the network, we choose on purpose to impose a constant and low load to the network whatever the number of nodes is. The outstanding result that this figure underlines is that SAFIR achieves high delivery ratio even when the number of nodes increases and not at the expense of a high control overhead (unlike ODMRP). Indeed, figure 4.15 shows the most important advantage of a hierarchical approach which is the control overhead saving. With the clustered approach, the flooding/prune process is limited within each cluster. Moreover, since in each cluster, the clusterhead is considered as the source of the multicast group, even if there are several sources for the same multicast group, only one flooding process is done. Figure 4.16 shows that, with the clustered approach, there is less data redundancy. Indeed, data are forwarded only to clusters where there are members. Moreover, the periodical data flooding done by each source of each multicast group is only done locally by each clusterhead once, whatever the number of sources is. One can expect that the packet delivery performance of SAFIR is better than the one of ODMRP since the overhead of SAFIR is significantly lower than the one of ODMRP. However, this statement is not true since the PDR measures the data packet losses which are due to collisions but also to route breakages. Since ODMRP presents redundancy, in this scenario where the number of members is important, packet losses due to route breaks are very low. Therefore, we can conclude that when the number of members is important, when ODMRP is employed, the packet losses are mainly due to collisions whereas when ODMRP is employed with SAFIR, packet losses are mainly due to routes breaks between clusters. This is confirmed through the results presented in figure 4.14 which presents the PDR as a function of the number of nodes when only 10% of the number of nodes are members. It shows that ODMRP suffers from more packets looses since the redundancy is less important whereas for the association SAFIR/ODMRP, the number of members has

Table 4.2 Interest of clustering: overview of the simulation scenario parameters

Scenario	1	2	3	4
Nb nodes	50	100	200	500
Nb members	20	40	80	200
Network size (m*m)	1000*1000	1000*2000	2000*2000	3100*3100

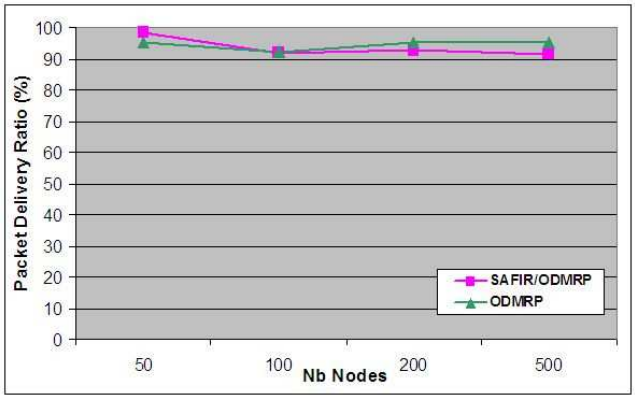

Figure 4.13 PDR as a function of the Number of Nodes with 40% of members

few influence. The performance of SAFIR can thus be improved if the clustering and the unicast routing protocols are enhanced to better face mobility.

4.4.4 About the influence of mobility

In these experiments, we evaluate the SAFIR protocol in mobile scenarios. Each node moves following the Random Waypoint mobility model with no pause time at a speed of 5 m/s. We compare the results obtained with this speed to the results obtained with a null speed. Figure 4.17 shows the PDR versus the number of groups in a 100 nodes scenario. There is one multicast group, with one data source node and 20 randomly chosen member nodes. SAFIR achieves a good delivery ratio even with mobility. The discrepancy between the scenario with and without mobility can be explained by the fact that the propagation of the data packets between clusters

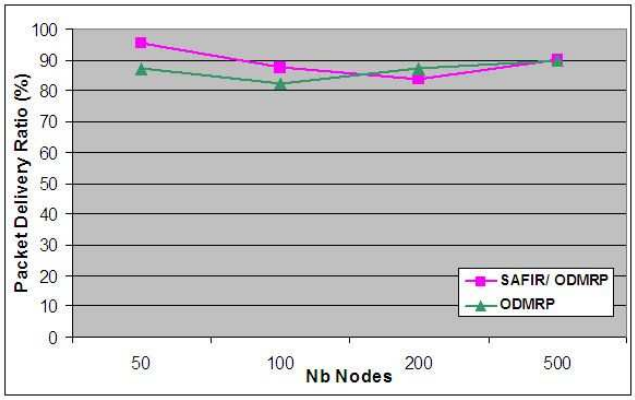

Figure 4.14 PDR as a function of the Number of Nodes - with 10% of members

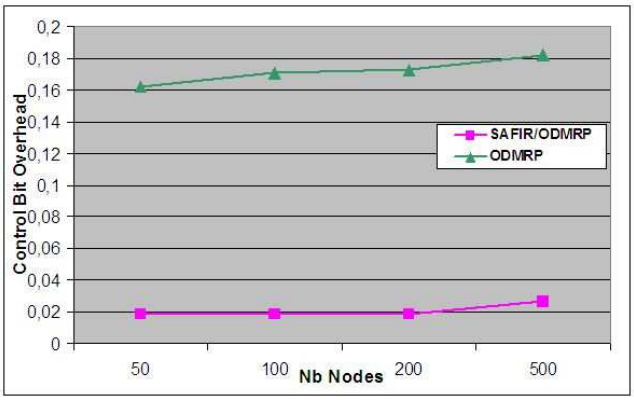

Figure 4.15 CBO as a function of the Number of Nodes

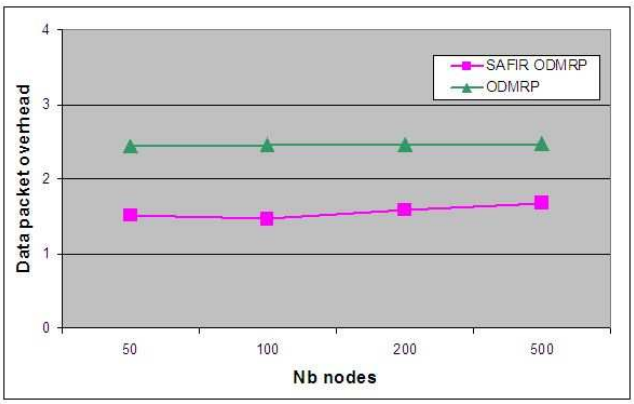

Figure 4.16 DPO as a function of the Number of Nodes

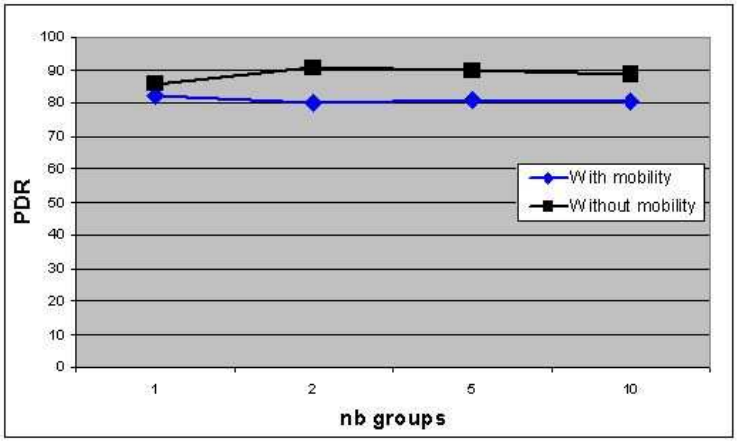

Figure 4.17 PDR as a function of the number of groups : Influence of the mobility on SAFIR performance

is done following a single path. Therefore, if a link breaks, data packets will be lost until the path is repaired. Nevertheless, we underline here that the performance of SAFIR with respect to mobility is dependent on the clustering protocol and also on the performance of the unicast routing protocol. Better robustness to mobility would have been observed if we had implemented the distance vector version of SAFIR. Nevertheless, due to the mobility profile of nodes in the tactical MANET which are supposed to move principally as groups, the mobility will have more influence on the intra-cluster multicast routing protocol than on the inter-cluster multicast routing protocol.

4.4.5 About the choice of the intra-cluster multicast routing protocol

In this last set of experiments, we evaluate the scalability of the SAFIR protocol with respect to the number of groups, the number of members per group and finally the number of sources per group. We compare results with the two well known categories of multicast routing protocols (mesh-based and tree-based) for the intra cluster multicast routing. For the mesh-based, we choose ODMRP and for the tree-based we choose STAMP. In these scenarios, the number of nodes in the network is fixed to 200, the traffic load is fixed to 5 pkts/s and nodes move following the Random Waypoint model with no pause time at a speed of 2m/s. Table 4.3 resumes the scenario used for our experiments. In all these experiments we observe that the PDR of both configurations are similar. The most interesting results are observed for the CBO and the DPO. As a reference, we also trace the results obtained with ODMRP without any clustering.

Figure 4.18 presents the relationship between the CBO and the number of groups in the network. As the number of groups increases, the control overhead of SAFIR/ODMRP increases. Indeed, for each group, the clusterhead of a cluster where there are members periodically performs flooding to refresh the mesh structure within the cluster. Since the number of received data packets is constant, it shows that as the number of groups increases, the control overhead becomes more and more

Table 4.3 Influence of the intra-cluster multicast protocol: overview of the simulation scenario parameters

	Nb of sources per group	Nb of multicast members	Nb of multicast groups
Scalability with Nb of sources	1, 2, 5, 10	80	1
Scalability with Nb of members	1	20, 40, 50, 100, 140	2
Scalability with Nb of groups	1	40	1, 2, 5, 10

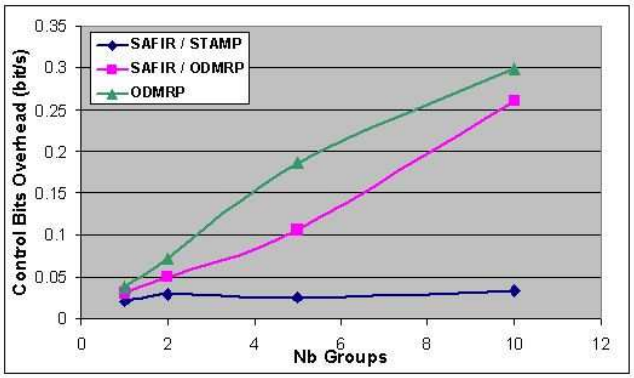

Figure 4.18 CBO as a function of the Number of Groups

significant with respect to the delivered data. In comparison, the control overhead of SAFIR/STAMP is low and therefore the increase is less significant. Indeed, STAMP relies on a shared-tree structure without any periodic flooding. Figure 4.20 presents an interesting result. It shows that the control overhead of the SAFIR/ODMRP association is independent from the number of sources whereas with ODMRP without any clustering it is directly linked to the number of sources. This fact can be explained by the SAFIR design. With SAFIR, in each cluster, the clusterhead becomes the source of the ODMRP mesh. Therefore, whatever the number of sources in the network is, each cluster will only perform periodical flooding once. This fact also explains why, in figure 4.21, the DPO of SAFIR/ODMRP does not increase when the number of sources increases. Therefore, the operation of SAFIR allows to mitigate one of the weakness of ODMRP, which is the dependency of the control and data overhead with respect to the number of sources. Thanks to SAFIR, ODMRP presents as good results as a shared-tree protocol (STAMP) with respect to the number of sources. Finally, figures 4.22 and 4.23 present the CBO and the DPO with respect to the percentage of member nodes i.e. the number of members. As expected, both CBO and DPO decrease when the number of members increases. Indeed, since more and more nodes are interested in receiving the multicast data, the number of nodes that are implied in the delivery of the datagram but are not member of the group decreases. This is confirmed by the fact that the DPO converges to 1 when the percentage of member nodes converges to 100% which means that almost all the nodes that forward the multicast datagram are also members of the multicast group. The assocation SAFIR/STAMP presents a better efficiency than the SAFIR/ODMRP association.

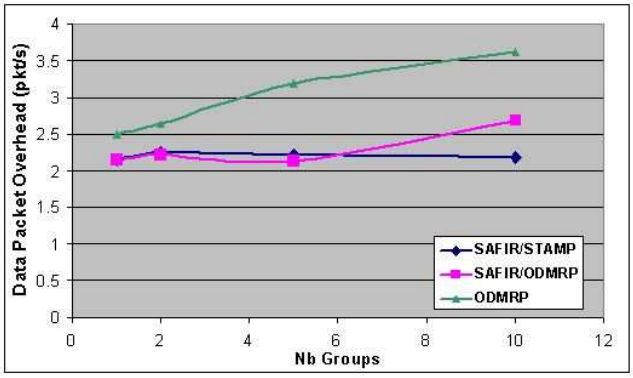

Figure 4.19 DPO as a function of the Number of Groups

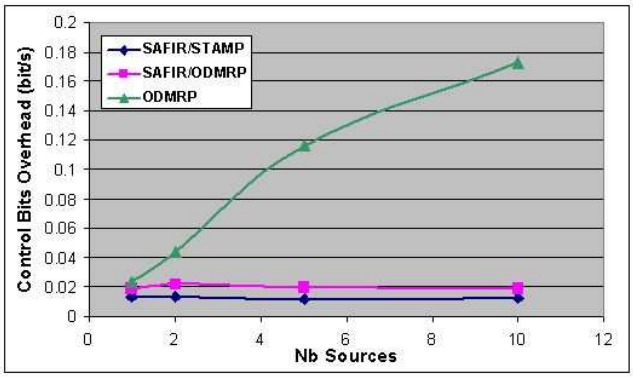

Figure 4.20 CBO as a function of the Number of Sources

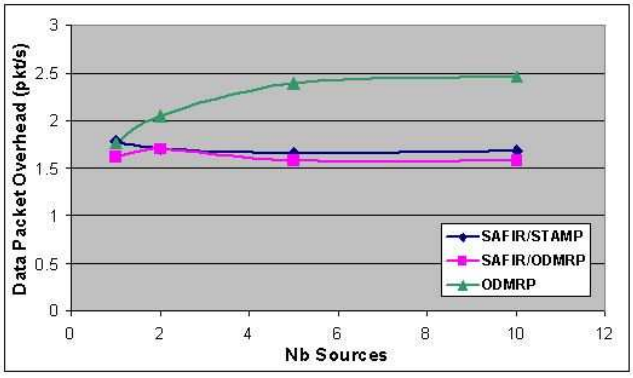

Figure 4.21 DPO as a function of the Number of Sources

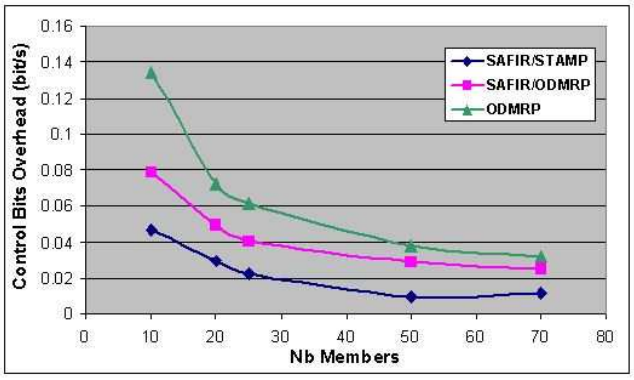

Figure 4.22 CBO as a function of the Number of Members

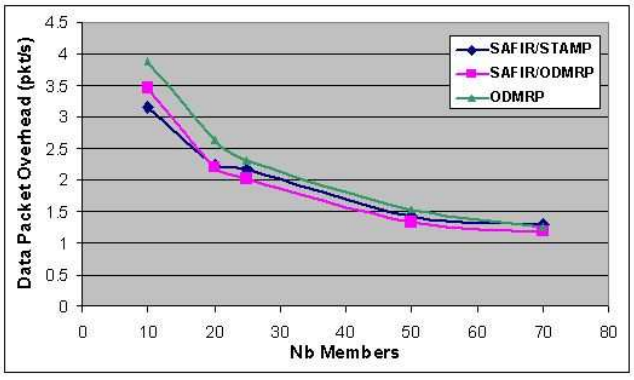

Figure 4.23 DPO as a function of the Number of Members

4.5 Conclusion

In this chapter, we present the ScAlable structure-Free Inter-cluster Multicast Routing (SAFIR) protocol as a solution for routing multicast information in a clustered, hierarchical network made of nodes that can be mobile. SAFIR is responsible for handling the inter-cluster multicast communications and assume that an intra-cluster multicast routing protocol such as STAMP is applied within each cluster. Therefore, the aim of the protocol is to define how a multicast datagram for a group G can be forwarded from cluster to cluster until reaching the clusters where the multicast members for the multicast group G are. Our protocol is optimized in term of efficiency (control and data overhead) since it benefits from the other services control messages to send the information needed for its operation. Moreover, SAFIR does not rely on any join/reply/leave messages to construct a multicast delivery structure on the cluster topology. The control information piggybacked in the control messages (the cluster multicast membership and the cluster topology information) are enough so that each clusterhead is able to decide on its own if it has to forward the multicast datagram received to other clusters, and if so to which clusters. We present the operation of the protocol in case the cluster topology information are distance-vector-based or link-state-based.

SAFIR is designed to operate in association with an intra-cluster multicast routing protocol that is responsible for delivering the multicast datagram within the cluster. We present in this chapter the way the two levels of protocols may interact to achieve seamless end-to-end communications within the tactical MANET. This interaction is presented in the general case, i.e. when SAFIR is employed with any flat or intra-cluster multicast routing protocol and also in the particular case of SAFIR employed with STAMP.

The performance evaluation of SAFIR demonstrates:

- the interest of the clustered architecture in term of control and data overhead saving. Indeed, we compare the efficiency of the ODMRP protocol in a clustered architecture employed in conjunction with SAFIR to the ODMRP protocol employed in a flat architecture. The results show an important saving in term of control and data overhead meanwhile there is no waste in term of packet delivery ratio.

- the protocol is resistant to mobility. Nevertheless, it is difficult to separate the responsibility of the clustering protocol, the responsibility of the unicast routing protocol and the responsibility of the multicast routing protocol in the robustness evaluation.

- that STAMP is a good choice for the intra cluster routing protocol compared to a mesh based protocol. Indeed, in almost all scenarios, the association SAFIR/STAMP presents the best results in term of efficiency.

The experiments we performed also demonstrate that the performance of SAFIR are dependent on the choice of the intra-cluster routing protocol. It would have been possible to have the same observation concerning the clustering or the unicast routing protocol if we had compared the results obtained with different clustering or unicast routing protocols than the one we choose. Thus, we reach there the limit

of the performance evaluation of a protocol (a multicast protocol in our case) when it is considered as a stand-alone protocol whereas it is closely dependent on the other protocols. Indeed, all the protocols that operate together to provide services to the network are closely dependent. The design of all the protocol and then the performance evaluation must be done in concert in order to optimize the performance of each protocol. It is what we have started to do by proposing to group the control information needed by SAFIR with the one needed by the clustering or the unicast routing protocol.

In the chapters 3 and 4, we present the protocols that can be deployed in the tactical MANET to achieve the scalability, robustness, efficiency and energy saving requirements imposed by the tactical environment. To provide a seamless end-to-end multicast service in the tactical network, the way these two protocols interact with the multicast protocols deployed in the networks connected to the tactical MANET must be defined. This work is presented in the following chapter.

Chapter 5

Interoperability of the Multicast Service in the Tactical Network

In the second chapter of this document, we present the tactical network structure. In this architecture, the tactical MANET is interconnected with several networks that we classify in three different categories with respect to structure they carry out for the multicast service i.e. Ethernet segments, local LANs and External IP Networks. We identify a "gateway node" as a node at the interface between the MANET and an IP External network. Each MANET node can be viewed as a Multicast Border Router, as defined in the RFC 2715 [121], since it runs two multicast components or routing protocols, one on the ad hoc network interface, and one on the wired network interface. We bring to the fore proxying as the best solution to interconnect wired IP networks through MANET. It means that to provide an efficient end-to-end seamless multicast service through the tactical MANET, the multicast service within the tactical MANET should be provided by the implementation of a MANET-specific multicast routing protocol. The two preceding chapters propose MANET-specific multicast routing protocols that address the requirements of the tactical MANET environment.

Employing MANET-specific multicast routing protocols raises interoperability issues since some translating and proxying mechanisms need to be defined so that the MANET-specific multicast routing protocol inter-operates with the traditional wired IP multicast protocols that will be employed in the LANs, in the Ethernet Segments and in the External IP Networks. This chapter proposes to address this interoperability issue which consists in answering the following question : how will the wired IP multicast solutions interact with the MANET multicast routing protocol to provide seamless end-to-end multicast connectivity? For instance, we will discuss how the multicast memberships can be exchanged between the different networks.

Some works have been done so far on interconnection of "hybrid MANET" i.e. mobile ad hoc networks connected to wired IP networks [4,88,109,110]. Nevertheless, in these works, the MANET is seen as an extension to the IP network where the mobile ad hoc network operates as stub network, meaning that all traffic carried by MANET nodes must either be sourced or sinked within the MANET. This is not compliant with the function of the tactical MANET which is a transit network which carries traffic entering and then leaving the network. This traffic is generated by hosts belonging to external IP networks, LANs or Ethernet Segment "attached" to the tactical MANET nodes.

In a first part, the tactical network structure is analyzed in order to identify precisely the issues that need to be addressed to provide seamless end-to-end multicast connectivity. Then, each issue is studied and solutions are proposed. These solutions are only at the proposition stage and need to be further studied. When proposing resolution to some issues identified in the first part, we consider two configurations for the multicast service in the MANET network, i.e. that STAMP is employed as a stand-alone multicast routing protocol or that it is employed in conjunction with the inter-cluster multicast routing protocol SAFIR.

5.1 Network structure analysis

A tactical MANET must provide interconnection between different types of network configurations. Due to the multicast model, these different types of networks are translated into multiple multicast-related protocols. Consequently, depending on

the location of the sources and the members on the different networks, the MANET nodes must interact with different protocols and must handle different information (e.g. multicast membership). Note that the addressing scheme is assumed to be consistent in the MANET network and the local LANs.

5.1.1 If the multicast actors belong to Ethernet segments

5.1.1.1 The case of sources

For workstations that are sources of a multicast group, the local ad hoc node acts as the local router. The ad hoc node connected to the local source node receives the multicast datagram and then acts as a source node within the MANET for the MANET-specific multicast routing protocol.

5.1.1.2 The case of members

For workstations that are members, the local ad hoc node acts as the local router. The workstations exchange IGMP messages with the local ad hoc node. When the ad hoc node receives a membership report message for a particular group, it must become a member for that group in the MANET. That way, the ad hoc node receives the multicast data traffic transiting in the MANET and then forwards it on its local LAN.

5.1.2 If the multicast actors belong to local LANs

5.1.2.1 The case of sources

The multicast data packets must reach the local ad hoc node, in order to be forwarded to other ad hoc nodes that have multicast members in their local LAN or to gateways. A solution must be proposed so that the traffic is forwarded from router to router on the local LAN to finally reach the local ad hoc node. This issue will be referred as issue 1.

5.1.2.2 The case of members

The workstations exchange IGMP messages with their FHMR on the local LAN, then the local LAN routers construct a multicast delivery structure within the LAN. To be able to forward the multicast data traffic (coming from another LAN or an external network) to local members, the data traffic must before be received by the local ad hoc node. Therefore, this local ad hoc node must be a member on the MANET network for every multicast group for which there are members on its local LAN. Consequently, a solution must be proposed so that the ad hoc node is aware of the local membership even if it is not directly connected to the workstations and therefore does not receive any IGMP messages. This issue will be referred as issue 2.

5.1.3 If the multicast actors belong to External IP Networks

5.1.3.1 The case of sources

If there are source nodes for a multicast group in an External IP Network, the multicast datagram initiated by such nodes must be forwarded to the gateway node in order to be then forwarded to the MANET nodes that have multicast members for that multicast group in their local LAN. Nevertheless, if there is not any member for the multicast group in the local LANs there is no need for such a behavior. The gateway node must therefore be a member node on the External IP Networks for every multicast group for which there are members on the local LANs. A solution must be proposed so that the gateway node is aware of the multicast membership of all nodes belonging to the tactical Internet. This issue will be referred as issue 3. The problem of gateway detection may also be considered (Issue 5). If there are several gateway nodes, the problem of the gateway duplication must be considered (Issue 6). Members

5.1.3.2 The case of members

If there are members of a multicast group in an External IP Network, the gateway node must forward any data traffic generated by a node on a local LAN or an Ethernet segment to the External IP Networks where the members are. To achieve such a behavior, a solution can be that the gateway node is aware of the multicast membership on the External IP Networks. Another solution can be that the gateway receives by default all the traffic generated by any local LAN. This will be referred as issue 4. Issues related to gateway duplication and gateway discovery must be considered. As long as gateway duplication is concerned, duplicate packet detection must also be considered.

5.1.4 Conclusion : Issues identification

The network analysis performed previously underlines several design goals that the multicast MANET nodes must achieve. These goals bring to the fore issues to solve. Figure 5.1 illustrates these different issues.

- The ad hoc node must be able to handle IGMP messages, which does not seem to present major difficulties.
- A solution must be proposed so that the data packets generated by a source on a local node are forwarded to the local ad hoc node in case the source is not directly connected to its local ad hoc node. **This point will be referred as Issue 1.**
- A solution must be proposed so that the local ad hoc node is aware of the local multicast memberships in case it cannot receive IGMP messages generated by members on its local LAN. **This point will be referred as Issue 2.**
- A solution must be proposed so that a gateway node is aware of the multicast memberships of all nodes in the tactical Internet (belonging to local LANs and Ethernet segments). **This point will be referred as Issue 3.** This solution

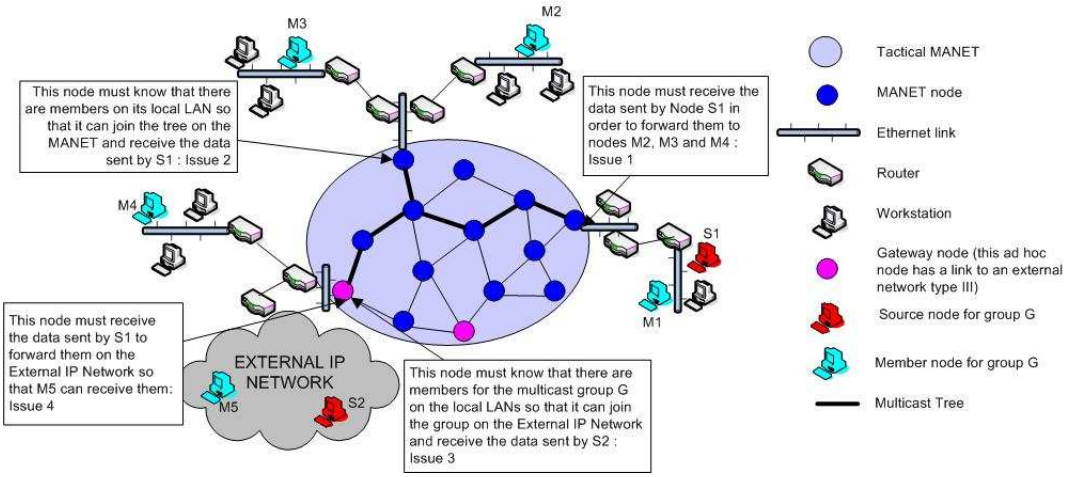

Figure 5.1 Illustration of the different issues to solve

must consider possible issues related to gateway detection **referred as Issue 5** and gateway duplication **referred as Issue 6**. In this last issue, duplicate packet detection will also be discussed.

- A solution must be proposed so that the gateway node receives any traffic generated within local LANs or Ethernet segments for multicast groups for which there are members on the External IP Network. Either, the gateway node is aware of the multicast memberships on the External IP Network or the gateway receives by default all traffic generated by any source on the local LANs or Ethernet segments. **This point will be referred as Issue 4.**

To solve these issues, we have to take into account a principal requirement which is to use IP multicast solutions on the External IP Networks without any modification in order to be fully compatible with any solution. We must be able to interface with the PIM-SM protocol as well as with the DVMRP protocol on the local LANs.

5.2 Issues resolution

5.2.1 Issue 1

The purpose of this part is to propose solutions to the following issue: how can an ad hoc node be aware of the local multicast source nodes that may exist in its local LAN? How can the traffic be received by the local ad hoc node?

5.2.1.1 PIM-SM as the local multicast routing protocol

If the multicast routing protocol used in the local LAN is PIM-SM, two solutions may be envisaged to solve this first issue.

The first solution is based on the behavior of the PIM Multicast Border Router[1]

[1]A PIM Multicast Border Router for a PIM-SM domain is a border router of the PIM-SM domain that speaks PIM-SM on some interfaces and speaks other multicast routing protocol on

(PMBR) described in the Appendix A of the RFC 4601 [40]. This appendix distinguishes two tasks for the PMBR. The task we are interesting in for this part is the one that ensures that the traffic from sources inside the PIM-SM domain reaches receivers outside the domain. There are two possible solutions for that:

- Another multicast component than the PIM-SM component of the PBMR is configured as a "wildcard receiver". In this case, the PIM-SM component of the PBMR must ensure that traffic from all internal sources reaches the PMBR until it is informed otherwise. To do so, the PBMR joins all active RPs in the PIM-SM domain. This causes all traffic in the domain to reach the PMBR. The PMBR may then act as if it were a DR with directly connected receivers and may trigger the transition to a shortest path tree.

- No other component is configured as a wildcard receiver. In this case, the PBMR must have explicit information as to which groups or (source, group) pairs the external domains wish to receive. If it has such information, the PMBR does not need to join all active RPs of the PIM-SM domain but only the RPs of multicast groups for which there are members on the external domains. However, the PMBR still need to act as a DR with directly connected receivers on behalf of the external receivers in being able to switch to the shortest path tree for internally reached sources.

In our network configuration, we are interested in the first solution since a tactical MANET node does not know the exact multicast membership of the "external networks", i.e. all other LANs and External IP Networks connected to other MANET nodes. Therefore each ad hoc node has to act as a wildcard receiver for external sources on the ad hoc multicast component in order to make the PIM-SM component join all the multicast groups.

A second solution can be envisaged if the protocol chosen for local LANs is PIM-SM. Indeed, if the local ad hoc node is set up to be the RP for all multicast groups on the LAN, each multicast source encapsulates its multicast data and sends it in unicast to the RP. This process is called the registering. Therefore, the ad hoc node is aware of the existence of the sources on its local LAN and receives the data packet they generate. It should be noted that in the normal operation of the PIM-SM protocol, the registering process is only temporary. Indeed, after a period of time, the RP sends back to the source a join message so that the source takes part of the tree and does not need any more to encapsulate its messages. Nevertheless, the data messages are still received by the RP. After this phase, called "register-stop"‘, some receivers may choose to switch to a source routed tree. Indeed, a shared tree does not optimize the forwarding path. Thus, if a receiver realizes that the route via the RP involves a significant detour compared with the shortest path from the source to itself, the receiver may send a join directly to the source to construct a source tree path. Nevertheless, even if such optimization is performed in the LANs, the data packet are still forwarded toward the RP.

other interfaces. Only one multicast routing protocol per interface is allowed. On each interface, the multicast routing protocol is run by a component, for example a PIM-SM component.

5.2.1.2 DVMRP as the local multicast routing protocol

If the multicast routing protocol employed on the local LAN is DVMRP, the local ad hoc node receives the data messages periodically flooded by the source to construct the source tree. Then, if the ad hoc node is configured to be a wildcard receiver on its wireless interface, it joins all multicast groups. This configuration makes the MANET node to receive all the multicast data messages.

5.2.1.3 Conclusion

Table 5.1 summarizes the different solutions for this first issue.

	Description	Applicability
Solution 1	The wireless interface of the MANET node is set as a wildcard receiver for all external sources	All IP multicast protocols
Solution 2	The multicast component on the wired interface of the MANET node is set as the RP for all multicast groups on the local LAN	Only if PIM-SM is chosen as the multicast routing protocol on the local LAN

Table 5.1 Potential solutions for the first issue

5.2.2 Issue 2

The purpose of this part is to propose solutions to the following issue: how can an ad hoc node be aware of the local multicast membership that may exist in its local LAN?

This issue arises when there are sources for a multicast group externally to the LAN and receivers for the same group in the LAN. A possible solution could be that every ad hoc node becomes a wildcard receiver on the LAN interface for all external sources. Thus, the ad hoc node would become a receiver for all multicast groups in the MANET and therefore would receive all data messages. Nevertheless, since each ad hoc node has potentially a LAN connected, such a solution implies that all ad hoc nodes are members of all multicast groups even if there is not any member on the local LAN. The multicast datagrams have to be flooded over the MANET. This solution is far from optimal and can therefore not be considered.

5.2.2.1 Domain Wide Multicast Group Membership Report

A protocol called Domain Wide Multicast Group Membership Reports (DWR) [43] was submitted to the IETF as an Internet draft in July 2000. This protocol proposes a solution so that group memberships inside a domain may be learned at the domain level by the border routers. In DWR, all border routers join a special multicast group (Domain-wide query multicast group) in order to perform the election of the Querier. This querier is responsible for sending periodic queries addressed to the domain-wide query multicast group. All routers in the domain must join the domain-wide query multicast group to be able to receive the query messages. When receiving a domain-wide query, a router sends back a domain-wide report to the domain-wide report

multicast address which includes the list of membership this router is aware of. All multicast routers are thus aware of the list of membership of all network routers.

The DWR protocol can therefore be a solution to the second issue. Indeed, if the local MANET node is the domain wide querier then, it will periodically send domain wide queries in its LAN to know the membership on this LAN. This protocol can be used whatever the multicast routing protocol used in the local LAN is. Nevertheless, the Internet Draft describing the DWR protocol has not been upgraded since 2000 and no implementation seems to exist.

5.2.2.2 PIM-SM as the local multicast routing protocol

Another solution can be proposed if the multicast routing protocol employed in the local LAN is PIM-SM. This solution is a configuration solution where the local MANET node is set as the RP for all multicast groups. Thus, the RP receives all join messages from the members and therefore is able to deduce the exact multicast membership in the local LAN.

5.2.2.3 Conclusion

Table 5.2 summarizes the different solutions for this second issue.

	Description	Applicability
Solution 1	The MANET node acts as the querier in the DWR protocol	This protocol has not been standardized and no implementation exists.
Solution 2	The multicast component on the wired interface of the MANET node is set as the RP for all multicast groups on the local LAN	Only if PIM-SM is chosen as the multicast routing protocol on the local LAN

Table 5.2 Potential solutions for the second issue

5.2.3 Issue 3

The purpose of this part is to propose solutions to the following issue: how can an ad hoc gateway node be aware of the multicast membership that may exist on all the local LANs of the tactical network?

The resolution of this issue is tightly related to the multicast routing protocol running on the multicast MANET. In this part we present a solution for the two protocols we propose in this thesis: Shared Tree Ad hoc Multicast routing Protocol (STAMP) and Scalable structure-Free Inter cluster Multicast Routing (SAFIR).

5.2.3.1 Resolution if STAMP is used as multicast routing protocol in the MANET

STAMP is a shared tree protocol where the first member for a multicast group becomes the core of the tree. One should remember that an ad hoc node is member for a multicast group if there is at least one node that is member for that group

in its local LAN. To announce itself to the remainder of the MANET nodes, a core periodically broadcasts a core announcement message. Therefore, if a node receives a core announcement message for a multicast group, it means that there is at least one MANET node that is member for that multicast group. Therefore, based on the core announcement messages received, a gateway node is able to determine for which groups there are members on the local LANs of the tactical network.

The membership status kept by the gateway is periodically refreshed each time a core announcement message is received. If after a determined period of time, no core announcement for a multicast group is received, it means that there is no member any more on the local LANs for the multicast group and that the entry has to be removed from the gateway multicast membership table. Therefore, each entry in the group membership status table kept by a gateway node is associated with a timer. The value of this timer is linked to the periodicity of the core announcement messages.

5.2.3.2 Resolution if SAFIR is used as multicast routing protocol in the MANET

SAFIR is a scalable multicast routing protocol for inter-cluster multicast communications. It is employed when the number of nodes in the network becomes so large that a traditional flat multicast routing protocol is not efficient. In SAFIR, each node belongs to a cluster where the clusterhead has the knowledge of the overall network membership. Indeed, the clusterhead of each cluster gathers the multicast memberships in its cluster and exchanges this information with the other clusterheads.

Therefore, in order that a gateway is aware of the multicast group membership of all local LANs, a special message of membership reporting should be defined to be exchanged between the gateway node and the clusterhead it depends on. There are two possibilities for these messages:

- it is the clusterhead that sends the message to the gateway in a unicast mode each time it detects a modification in its multicast membership information;
- it is the gateway that periodically queries its clusterhead.

The first approach needs that the clusterhead gets the knowledge of the gateway node belonging to its cluster whereas the second approach does not need such knowledge. However, with the second approach the gateway sends periodical messages even if the membership information has not changed. Moreover, the second approach is a two-phase solution i.e. a "query-reply" solution, whereas the first approach is a one-phase solution.

5.2.3.3 Conclusion

Table 5.3 summarizes the different solutions for this third issue.

5.2.4 Issue 4

The purpose of this part is to propose solutions to the following issue: how can a gateway node receive any traffic generated by sources on local LANs for multicast

	Description	Applicability
Solution 1	The gateway node knows the ad hoc network multicast membership thanks to the received core announcements	If STAMP is employed as the multicast routing protocol in the tactical MANET.
Solution 2	The clusterhead sends to the gateway the ad hoc network multicast membership each time it detects a modification. It is a one phase operation and the clusterhead needs to know the gateway	For clustered networks, i.e. if SAFIR is employed as the inter cluster multicast routing protocol in the tactical MANET.
Solution 3	The gateway periodically asks its clusterhead for the ad hoc networks multicast membership. It is a two phases operation that is done periodically even if there is no changes	For clustered networks, i.e. if SAFIR is employed as the inter cluster multicast routing protocol in the tactical MANET.

Table 5.3 Potential solutions for the third issue

group for which there are members on the IP External Network?

Two solutions may be considered. The first one relies on the assumption that the gateway node is aware of the multicast memberships of the External IP Networks. Therefore, it may become a member on the ad hoc network for all multicast groups for which there are members on the External IP Network. This solution assumes that an inter-domain multicast membership advertisement protocol such as BGMP is employed in the External IP Network. Unfortunately, we cannot assume or impose such an assumption on these types of networks. Therefore another solution may be considered.

The second solution relies on the fact that a gateway node receives all the data traffic generated within the local LAN. Thus, the gateway node acts as if it were a DR of a local LAN and if its local LAN sends it all the multicast data to forward on the network. To receive all the traffic, there are two solutions:

- Each local ad hoc node that gets multicast data from its local LAN unicasts it to the gateway whether or not there are members on the ad hoc network.
- The gateway becomes a member for each multicast group regardless of whether there are members on the ad hoc network or not.

Since the first approach is not efficient, we will only consider the second solution. This solution is equivalent to the configuration in which the gateway node is a wildcard receiver for all external sources on its wired interface. In this part we present a solution for two ad hoc multicast routing protocols: STAMP and SAFIR.

5.2.4.1 Resolution if STAMP is used as multicast routing protocol in the MANET

In STAMP, the first node that joins a multicast group becomes the core node for the multicast group. Therefore, since we do not want the gateway to be the core node for each group, a gateway joins a multicast group only when it gets a core announcement for the group.

One may ask what happens if it exists multicast groups for which there are sources in the LANs and no member. Indeed, in such a situation, no core announcement will be sent in the MANET and the gateway cannot join the group. The ad hoc node "source" is responsible for detecting such a situation. Thus, when an ad hoc node receives multicast data, two situations may happen:

- The node knows any core node address for that group. It means that it exists at least one member for that group and that the gateway has joined the multicast tree. The multicast data packets are forwarded to the core following the normal STAMP behavior.

- The node does not know the core node address for that group. It means that there is not any member for that group in the MANET. Therefore, the node unicasts the multicast data to the gateway node.

5.2.4.2 Resolution if SAFIR is used as multicast routing protocol in the MANET

In SAFIR, it is the clusterhead that gathers the multicast membership information in each cluster. Therefore, a gateway node sends a join_all_multicast_groups message to its clusterhead following the intra-cluster multicast routing protocol behavior. The intra-cluster multicast routing protocol must be enriched so that it can handle these join_all_multicast_groups messages. This information is shared by all clusterheads. Thus, when an ad hoc node receives a multicast data packet from its local LANs, it forwards it within its cluster following the intra-cluster multicast routing protocol. When the data packet reaches the clusterhead, it is forwarded to the cluster the gateway belongs to following the SAFIR operation. Finally, the clusterhead gives the responsibility for forwarding the data to the intra-cluster multicast routing protocol. Since the gateway is a member of all groups, it receives the multicast datagrams.

5.2.4.3 Conclusion

Table 5.4 summarizes the different solutions for this fourth issue.

	Description	Applicability
Solution 1	The gateway node joins all multicast groups for which it receives a core announcement. If a node that receives multicast data from its local LAN does not know a core for the group, it unicasts the data to the gateway node.	If STAMP is employed as the multicast routing protocol in the tactical MANET.
Solution 2	The gateway node sends a join_all_multicast_groups message to its clusterhead meaning it is a member for all multicast groups.	For clustered networks i.e. if SAFIR is employed as the inter cluster multicast routing protocol in the tactical MANET.

Table 5.4 Potential solutions for the fourth issue

5.2.5 Issue 5

Until now, we have considered that the gateway node(s) is (are) known to the other ad hoc nodes. In the usual ad hoc network operation, mobile nodes must detect available gateways through a gateway discovery mechanism. One can envisage that in a tactical MANET, gateway nodes are predefined before the mission, and that this information is given to all mobile nodes before deploying. In such situations, a gateway discovery mechanism is not needed. Nevertheless, we must envisage deployments where the gateway nodes are not known in advance.

Some work has been done so far in the research community on gateway discovery in hybrid ad hoc networks. We can distinguish three approaches i.e. the proactive, reactive and hybrid approaches. Proactive and reactive approaches present the same behavior as for ad hoc routing protocols:

- Proactive discovery: all gateways periodically broadcast their IP address (and their services) throughout the MANET.

- Reactive discovery [62, 128]: A mobile node that wants to know the gateway address broadcasts a message throughout the MANET soliciting a connection to the External IP Networks. A gateway receiving this message will reply back to the mobile node.

A third solution, the hybrid discovery approach [75, 104], has been proposed as a trade off between the advantages of proactive and reactive approaches. In such an hybrid approach, the overhead costs are reduced thanks to a limit-scoping broadcast of the periodical advertisements. Indeed, the periodic advertisements of the gateway are not flooded to the whole network but are only sent to the mobiles nodes that are at maximum n hops from the gateway, where n is defined thanks to the TTL of the advertisement messages. The mobile nodes that are more than n-hops away from the gateway must solicit advertisement reactively.

Some works propose that the gateway discovery mechanism is integrated in the MANET routing protocol or in the neighbor discovery protocol [107].

Performance evaluations of the different approaches have been performed [50,54]. These papers show that dealing with the average delay for a node to set up a route to the Internet, proactive approaches present better delay than hybrid ones which are better than reactive approaches. In term of control overhead, the reactive approaches generate less overhead as long as the number of nodes that needs to know the gateway address remains low. These conclusions are naturally the same as with unicast routing protocols.

In our context, it seems that the proactive approaches better suit our constraints. Indeed, all MANET nodes may need to know the gateway address. Moreover, the unicast routing protocol employed in our network is a proactive protocol. Therefore, to limit the additional overhead, unicast control messages can be reused to convey the gateway identity throughout the network.

Several solutions may be considered to carry out this behavior:

- The gateway nodes broadcast a special gateway_advertisement message periodically.

- Hello messages can be used to advertise the gateway address as described by Rosenschon [107]. In this paper, they propose to enhance the AODV protocol so that a gateway node sets a flag (the I-flag) in the HELLO header to mark its HELLO messages as originated by a gateway. Thus, nodes that are neighbors to the gateway are aware of that gateway. In a further step, the gateway neighbors set again the flag to indicate that their HELLO message contain gateway information and include the gateway address in an unused field of the HELLO header. In this approach, only one gateway can be advertised.

- The OLSR RFC proposes the Host and Network Association messages. These messages are sent by nodes that have a non-OLSR interface to advertise the network addresses they can reach through these interfaces. In the RFC, a node that has a non-OLSR interface is supposed to be a gateway. In our configuration, each ad hoc node may have a non-OLSR interface, but we do not want each node to be a gateway and therefore to send HNA messages. Therefore, these HNA messages can be a solution for the gateway advertisement only if there are authorized on the gateway nodes.

- Another proposition with the OLSR protocol: Hello messages can be used by the gateway to inform its neighbor of its gateway status. Then, if OLSR is used, it can be the MPR of the gateway (or the gateway itself if it is an MPR), that indicates in its TC messages the address of the gateway. Since the TC messages are received by all nodes in the network, all nodes may be aware of the gateway address. This approach allows advertising several gateway addresses.

- If a clustering protocol is employed, any of these preceding propositions can be employed with a scope limited to the cluster range. The clusterheads are responsible for exchanging the gateway address information among them.

Table 5.5 summarizes the different solutions for this fifth issue.

	Description	Applicability
Solution 1	The gateway node sends periodical advertisements	
Solution 2	The gateway node set a flag in its Hello messages to indicate that it is a gateway	Only if the AODV protocol is employed as the unicast routing protocol in the MANET.
Solution 3	The gateway sends HNA messages to the MANET network	Only if the OLSR protocol is employed as the unicast routing protocol in the MANET. The HNA capability must be turn off on all ad hoc nodes except the gateway
Solution 4	The TC messages can be modified to include the gateway address.	Only if the OLSR protocol is employed as the unicast routing protocol in the MANET.

Table 5.5 Potential solutions for the fifth issue

5.2.6 Issue 6

Until now, we have only considered that the ad hoc network is connected to the External IP Networks through a single gateway. However, it may be possible that a tactical MANET is connected to the External IP Networks through different gateways. If multiple gateways exist, several issues may occur:

- Multicast traffic going out of the MANET to the External IP Networks may be duplicated
- Multicast traffic entering the MANET may be duplicated, leading to overhead or detection of packet duplication issues.

Therefore, similarly to a LAN where a DR needs to be designated, we propose to elect only one gateway among the possible gateways. The election process must be distributed and must lead to the same choice on all ad hoc nodes. Thus, the distance to the gateway cannot be a criterion since the distance to the gateway will not be the same for all ad hoc nodes. We propose to use the address or the id of the gateway as the criterion. For example, the one which has the lowest address on the MANET is the elected gateway. Therefore, only one gateway node is allowed to inject traffic within the MANET and to forward traffic out of the MANET. This allows to minimize the duplicate packet detection issue. Nevertheless, some mechanisms for duplicate packet detection (DPD) still need to be implemented for STAMP and SAFIR operation. This issue is currenlty under discussion at the MANET WG of the IETF concerning the Simplified Multicast Forwarding protocol [83]. Two approaches are considered: the header content identification based (I-DPD) and secondly, the Hash based duplicate detection (H-DPD). For the first approach, packets are identified thanks to explicit identifiers from the IP header. The field "identification" of the IPv4 header is proposed to be used as the key identifier. Nevertheless, the IPv4 header identification value is not always generated properly. Consequenlty, the hashing duplicate packet detection based approach is proposed. This solution applies an MD5 hash of the non-variant header fields, option fields and data content of the IPv4 multicast packet resulting in a 128 bits value. Charles Perkins proposes in an email to the MANET mailing list to employ the SHA-1 protocol on the source IP address and the identification field, resulting in a 16 bit value. One of these two mechanisms can therefore be implemented by each node for duplicate packet detection purpose.

Note that since each node knows all possible gateways, some QoS features such as load balancing depending on the multicast group address or election depending on QoS criterion may also be employed as long as the choice of the criterion leads to a common gateway election on all nodes. The elected gateway will go on advertising its address periodically whereas the other potential gateways will stop. Therefore, if the elected gateway stops sending its advertisement messages, the other potential gateways can detect that it is no more available and can re-start the electing process.

5.2.7 Conclusion

Table A.2 summarizes the different solutions proposed to solve issues 1 and 2. Table A.3 summarizes the different solutions proposed to solve issues 3 to 6.

Solution description	Applicable if the multicast protocol employed in LAN is	Resolution of issue number	
		1	2
The local MANET node is set as the RP	PIM-SM	**X**	**X**
The ad hoc interface is set as a wildcard receiver for all external sources	All	**X**	
The ad hoc node acts as the querier in the DWR protocol	All		**X**

Table 5.6 Solutions for the issue 1 and 2

5.3 Conclusion

Within the tactical network architecture, the tactical MANET is a special type of hybrid MANET where the MANET is not a stub network but a transit network. Whereas in commercial use of mobile ad hoc network, the wireless network is seen as an extension of a wired IP network operation as a stub, its use in the tactical network structure places it as a transit network carrying traffic entering and then leaving the network. As long as the multicast service is concerned, this particularity raises the issue of the interoperability between the multicast routing protocols employed in the wired IP networks and the MANET-specific multicast routing protocols. In this chapter, we address this interoperability problem considering firstly the structure of the network from a multicast point of view in order to identify the functionalities needed to provide end-to-end seamless multicast connectivity through the tactical MANET.

In this first part, we study the repercussions on the multicast flows of the repartition of the different multicast actors among the different types of the wired IP networks. For instance, we consider what happens if a source node of a multicast group is located on a local LANs whereas members of this same group belongs to External IP Networks. What information needs to be exchanged on the local LANs, on the tactical MANET so that the multicast flows can be routed from the source to the destinations? What is the responsibility of the local MANET node with respect to the source node? What are the responsibilities of the local MANET nodes and the gateway nodes within the MANET with respect to the multicast control information and the multicast data flows ? This first phase brings out six problems that need to be solved to achieve the end-to-end multicast connectivity objective. In a second part, we propose some solutions to the different issues identified previously. These solutions are only at the proposition stage and need to be further studied. When proposing solutions to some of these issues, we consider two configurations for the multicast routing within the MANET, the use of STAMP and the use of SAFIR. Additional work needs to be done on this part, for example, performance evaluation of the different propositions has not been performed. Such an evaluation would allow to leverage a solution when several resolutions are proposed.

Solution description	Applicable if the multicast protocol employed in MANET is	Issue number			
		3	4	5	6
The gateway node knows the ad hoc network multicast membership thanks to the received core announcements	STAMP	**X**			
The clusterhead sends to the gateway the ad hoc network multicast membership each time it detects a modification.	SAFIR	**X**			
The gateway periodically asks its clusterhead for the ad hoc networks multicast membership	SAFIR	**X**			
The gateway node joins all multicast groups for which it receives a core announcement. If a node that receives multicast data from its local LAN does not know a core for the group, it unicasts the data to the gateway node.	STAMP		**X**		
The gateway node sends a join_all_multicast_groups message to its clusterhead meaning it is a member for all multicast groups.	SAFIR		**X**		
The gateway node sends periodical advertisements	Any			**X**	
The gateway node set a flag in its Hello messages to indicate that it is a gateway.	Any and AODV as unicast			**X**	
The gateway sends HNA messages to the MANET network	Any and OLSR as unicast			**X**	
The TC messages can be modified to include the gateway address.	Any and OLSR as unicast			**X**	
The node id or address is used to choose the best gateway	Any				**X**

Table 5.7 Solutions for the issue 3 to 6

Chapter 6

What About Unicast Scalability ?

This thesis mainly focus on the multicast service in a tactical network. In the second chapter, we examine the architecture of the multicast routing service and we choose clustering as the solution to provide scalability in the tactical environment. Therefore, through chapters 3 to 5, we study how to provide a multicast service in a clustered MANET. However, the majority of the communications in a tactical MANET are point-to-point communications. Therefore, a MANET node must implement a unicast routing protocol. Moreover, the multicast routing protocol we propose rely on an underlying unicast routing protocol. Therefore, in this last part, we are going to consider the unicast routing service in a clustered MANET. After a review of the literature concerning unicast routing in MANET, it appears that the study of scalable unicast routing relying on a clustered network is only partially completed.

Unicast routing in MANET is an area of great interest in the research community as illustrated by the important amount of submitted Internet Drafts. The IETF MANET WG [84] has studied two classes of unicast routing protocols, the proactive routing protocols and the reactive routing protocols. The reactive approach is based on discovering routes at the time an application needs to communicate to another node. The protocol uses control messaging (usually a controlled broadcast technique) to discover the portion of network topology that is relevant for routing the specific application datagram to the destination node. This type of approach allows the overhead to scale with the load of the network. However, due to the initial route discovery that must occur, initial delays are experienced, and may become a limiting factor for some types of application traffic. Moreover, when the number of communication pairs and the pair dynamics increase, the overhead becomes too important. The proactive approach is based on periodical exchanges of control information between routers that allow topology to be discovered and routing tables to be built and maintained continuously. The routes are then available when needed to forward datagrams. The overhead associated with proactive approaches is dependent on the desired settling time of routes with respect to mobility events, and is independent of network load. Control overhead is the main drawback of proactive solutions. This makes proactive protocols poorly scalable when the size of the network increases in term of number of nodes. The generating overhead may become so important that it consumes the whole bandwidth and then prevents any traffic to be transmitted. Moreover, storing routes to every node of the network may become unfordable in large scale networks with memory-constrained nodes.

OLSR (Optimized Link State Routing Protocol [28]) is a well-known link state proactive routing protocol commonly employed in the deployed MANET systems. In a link state routing protocol, each node senses its neighborhood by periodic exchanges of control messages (HELLO messages in OLSR). That way, each node learns its local vicinity and the status of the links it has with its neighbors (unidirectional or bidirectional). Then, this information is periodically advertised in the whole network thanks to control messages called TC (Topology Control) messages in OLSR. This allows nodes to compute routes to any potential destination node in the network. Additionally to this traditional link state operating, OLSR uses the concept of MultiPoint Relay (MPR) [61] in order to optimize the flooding of control messages in the network as illustrated by figure 6.1. Nodes selected as MPR are the only one allowed to forward the TC messages they receive. The nodes, which are selected

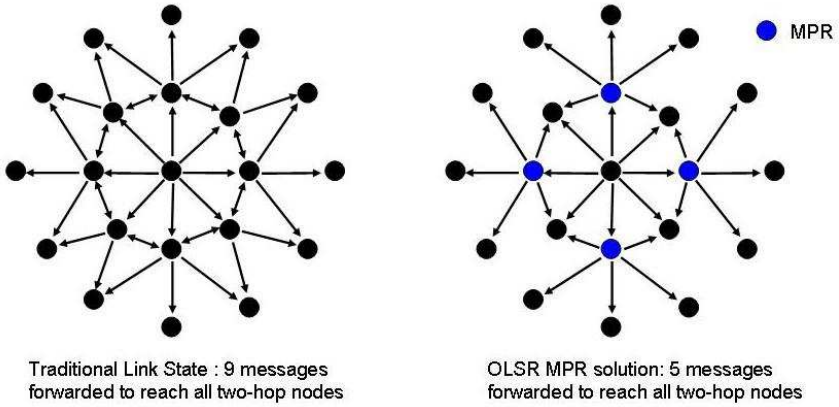

Figure 6.1 Illustration of the optimized MPR forwarding process

as MPR by some neighbor nodes, periodically announce this information in their control messages. Thereby, a node announces to the network that it has reachability to the nodes which have selected it as MPR. In route computation, nodes selected as MPRs are used to form the minimal route from a given node to any destination in the network. Thanks to the MPR concept, OLSR proposes three optimizations. Firstly, the nodes selected as MPRs are the only nodes allowed to forward the TC messages; therefore there are fewer nodes participating in the TC broadcasting. Secondly, the TC messages are only generated by nodes that have been selected as MPR; the number of TC messages sent is thus reduced. Thirdly, the size of the TC messages is decreased since a node selected as an MPR only advertises the links to the nodes having selecting it as MPR rather than the links with all its neighbors.

We remind here the basic definition of a node selected as MPR by a neighbor. Each node in the MANET selects its MPR following these two rules:

- Any 2-hop neighbor must be accessed by at least one MPR;
- The number of MPR should be as few as possible.

Therefore, the MPR concept allows OLSR to reduce both the number of broadcast packet transmission and the size of the control packets, leading to an efficient flooding of control messages in dense networks.

Even if OLSR allows reducing the control overhead in dense networks, it still presents scalability issues:

- When the network is not dense (i.e. sparse), all neighbors are chosen as MPR and therefore the protocol turns into to a pure link state protocol which generates a large amount of control overhead.
- Each node must store a route to all other nodes in the network which may be unfordable for low capacity devices.
- In case of link breakage, the link recovery impacts all nodes in the network generating an important and useless overhead.

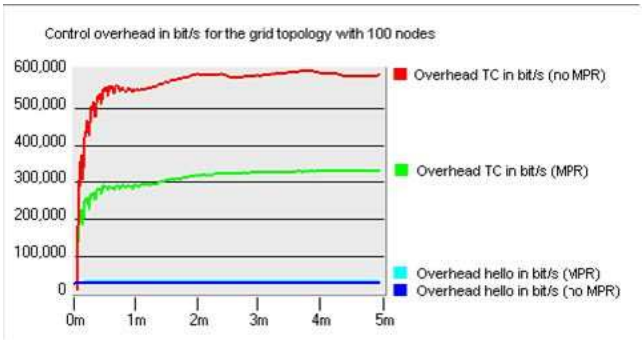

Figure 6.2 Comparison of the Hello and the TC overhead [73]

- The overhead of control messages is directly linked to the number of nodes since each TC message must be forwarded to the entire network. Thus when the size of the network increases in term of number of nodes, the control overhead may jeopardize the overall network performance.

In this chapter, we are going to review, in a first part, previous works that focus on improving the scalability of the OLSR protocol. A solution to adapt the well known OLSR unicast routing protocol to a clustered mobile ad hoc network is presented in a second part. Finally, the theoretical and practical analyses of this solution are provided in the third part.

6.1 State-of-the-art of scalable enhancements of OLSR

OLSR is a protocol that has been widely evaluated through theoretical analyses and simulations studies. One interesting result of these works is that the overhead caused by the TC messages gives the major contribution of the control overhead of OLSR. The figure 6.2 extracted from [73] illustrates this result. Consequently, it is not surprising to see that the protocols proposed to enhance the scalability of OLSR focus on reducing the overhead caused by the TC messages either by reducing their size and/or their number.

In [27], the author proposes to integrate the Fisheye routing [99] inspired from the Fisheye concept [68] into OLSR. The Fisheye routing consists in adapting the frequency of the forwarding of topology information to the distance to the source. Thus, nearby nodes receive topology information more frequently than farther nodes. The frequency of topology information updates decreases as the distance increases. The idea behind this optimization is that a node does not have to know the exact routing information about far destinations. A vague idea of the node location is enough to forward the data in the right direction. As the packet goes closer to the destination, the routing information becomes more and more accurate and the packet is routed more precisely. A simulation study [2] shows that Fisheye OLSR greatly improves the scalability of OLSR. Nevertheless, the problems of storage and control overhead that make OLSR poorly scalable are not entirely solved. Each node still has to store and compute a route to all potential destination nodes in the network.

Moreover, as the number of nodes increases, even if the periodicity is reduced, the TC messages still have to be broadcasted over the whole network. This point may present performance issues. Indeed, when the number of messages to be forwarded increases, the number of collisions and thus lost messages also increases which may cause route errors due to faults in the routing table. Moreover, the loss probability increases as the messages have to be propagated farther. Finally, a link breakage on a part of the network still has a global impact on the overall network since the control information must be updated in each node routing table.

To overcome these issues, the solution is to limit each node's view of the network by aggregating nodes. Indeed, aggregation enables the reduction of the algorithm complexity and the optimization of the resources (e.g. memory, medium, etc.) and simplifies the network management. In MANET, the aggregation of nodes is performed thanks to the clustering technique. Clustering allows to introduce levels of hierarchy in the network. Having levels of hierarchy enables the growth of the size of the each node's routing table to be only logarithmic instead of linear with respect to the number of nodes in the network. Based on a clustering protocol, several propositions to enhance the OLSR protocol have been made.

The first one is the Hierarchical OLSR protocol (H-OLSR [48]). H-OLSR creates a hierarchical topology assuming that some nodes in the network have better communication capabilities in term of radio propagation range, number of interfaces, frequency bands, battery life ... For instance, a node at the level one has only one interface whereas nodes at the level two have two interfaces, one to communicate with nodes at level one and one to communicate with nodes at level two. This second interface must have a longer transmission range. If nodes have the necessary communications capabilities, more levels of hierarchy can be built. Thus, nodes with better capabilities become clusterheads for nodes at lower levels. If more than two levels are established, a clusterhead of a node at level N-1 is considered as a simple node for nodes at level N+1. Topology information are sent only within the cluster and clusterheads at a same level exchange addresses of their local nodes through direct communications. Nodes of a cluster have enough information to route traffic to any same-level same-cluster destination. To reach any other destinations (other levels and/or other clusters), the traffic must be firstly routed through the clusterhead and then forwarded to the appropriate same-level clusterhead. This may lead to suboptimal paths when, for example, the source and the destination are close but belong to different clusters. Moreover, the assumption of the existence of higher capabilities nodes is a strong assumption that may not be verified in tactical MANET.

OLSR tree [9] defines a clustering algorithm to introduce hierarchy in OLSR without assuming heterogeneity of the network nodes. The clustering algorithm is based on the connectivity of nodes (the number of one-hop neighbors). The network is divided into trees, where the root of the tree, the clusterhead, is the node having the maximum local connectivity. Once trees are created, a maintenance process is run to adapt the structure to the topological changes. A hierarchical routing protocol based on OLSR is then employed. Routing within the tree scope is done with OLSR as if there were no tree. To route to other trees, OLSR is applied on the cluster topology thanks to "super messages" (Super TC, Super Hello, ...) exchanged by clusterheads. When a node needs to send data to a node outside its tree, it first sends the traffic to its root which then forwards the traffic to the destination node following the cluster

path. This may overload the clusterheads and produce suboptimal paths. OLSR tree proposes an interesting approach to improve OLSR scalability. Nevertheless, it is dependent on the clustering algorithm which itself is based on connectivity, i.e. a dynamic parameter in mobile networks. Consequently, cluster topology stability may be poor. Moreover, applying OLSR on top of the cluster topology may generate superfluous overhead.

The C-OLSR protocol has been presented recently [106] and proposes a modification of OLSR which makes use of clustering to reduce the protocol overhead. Contrary to OLSR Tree, the protocol does not depend on a defined clustering protocol but assumes merely that a clustering algorithm is being executed in the ad hoc network. C-OLSR uses regular OLSR inside every cluster and TC messages forwarding is thus limited within the scope of a cluster. Then, the authors choose to leverage the same mechanisms of plain OLSR to the level of clusters. Therefore, they define new C-Hello and C-TC messages to emulate the behavior of an OLSR node by a cluster. C-MPR clusters are elected thanks to the C-Hello messages. Three propositions are made with respect to which node in a cluster is responsible for the generation of the cluster topology messages (C-Hello and C-TC). It can be the clusterheads that generate both the C-Hello and the C-TC messages or the border nodes [1] or an hybrid solution where the border nodes generate the C-Hello messages and the clusterheads generate the C-TC messages. The C-Hello messages must be forwarded over the entire neighbor clusters so that each node of a cluster may compute its own C-MPR set and the C-TC messages must also be forwarded over the entire clusters that are selected as C-MPR clusters.

As in OLSR Tree, applying OLSR at the cluster level imposes to exchange Hello-like (Super-Hello or C-Hello) and TC-like (Super-TC or C-TC) messages which may generate an important overhead. Moreover, in case of loss of one of these messages, the integrity of the routing function may be jeopardized. We propose a solution to adapt OLSR to a clusterhead environment where regular OLSR is applied inside every cluster for intra-cluster communications but where inter-cluster communications do not rely on a version of OLSR at the cluster level contrary to what is done in both OLSR Tree and C-OLSR. The remainder of the chapter is organized as follows. Firstly, a description of our protocol is proposed. Then, theoretical and practical analyses of the protocol overhead compared to the Fisheye OLSR and the C-OLSR solutions are provided.

6.2 Protocol description

6.2.1 Overview

We propose a routing protocol based on OLSR which aims at improving the scalability features of the OLSR protocol in large-scale ad hoc networks. Our protocol uses clustering to greatly reduce the topology overhead and the routing table size. The routing protocol is fully independent of the clustering protocol used. We only assume that a clustering protocol is being executed in the ad hoc network. As long as the intra-cluster communications are concerned, the basic OLSR is used in every cluster

[1] a border node is a node that has a neighbor node belonging to a different cluster than its.

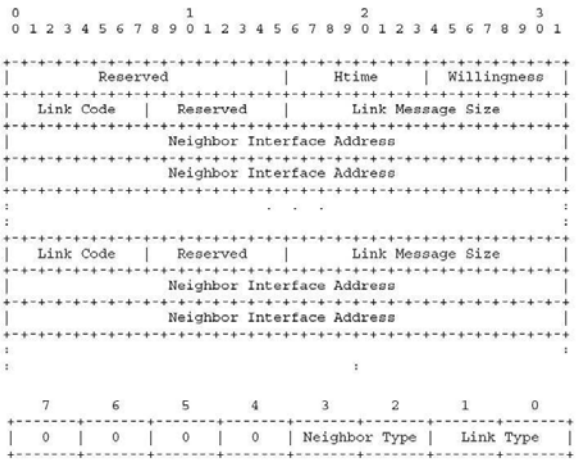

Figure 6.3 Hello message and Link Code field format

and the propagation of the topology control information is limited within the cluster. For out-of-cluster routing purposes, contrary to other OLSR-based approaches which rely on clustering such as OLSR-Tree, a version of OLSR is not applied on the clusterhead topology. Indeed, rather than applying the complex OLSR message exchange and MPR selection on clusterheads, in our solution clusterheads are only required to send special TC_Cluster messages over the network. Thanks to the information contained in these TC_Cluster messages, when a node wants to send data to a destination not belonging to its cluster, the sender knows the next hop node toward the clusterhead the destination depends on. This is the only information that each node must have to be able to route data packets. Regarding the clustering protocol, employing a K-hop clustering algorithm which forms clusters with diameter larger than 2 hops is recommended. We also assume that every node is aware of its clusterhead address.

6.2.2 Hello messages

The first modification we perform on the OLSR protocol deals with the Hello messages. Each node must include its clusterhead membership information, i.e. the address of its clusterhead, in its Hello message so that its neighbors may know if they belong to the same cluster. Hello message format and link code format are defined in the RFC 3626 (cf. figure 6.3). With the aim to be compliant with the regular OLSR protocol (i.e. we want that our protocol can be used in networks where both "regular" OLSR and our protocol co-exist), the Hello message format is not modified. An Hello message is made of several blocks, one block per Link Code value. The link code specifies information about the link between the interface of the sender and the following list of neighbor interfaces. It also specifies information about the status of the neighbors. We propose to add a new Link Code to indicate that the clusterhead address is advertised. Consequently, in an Hello message sent by a node advertising its clusterhead address, one of the link code blocks will be dedicated to the clusterhead address of the node sending the Hello. A link code is made of two parts of two bits each, the Neighbor Type and the Link Type. The four possible

values for the Link Type are already defined. For the Neighbor Type field, three of the four possible values are used. One for a symmetrical neighbor, one for an MPR neighbor and one for a non symmetrical neighbor. Therefore, we propose to use the last available value of the Neighbor Type field to add a new value, CH_NEIGH. The new link code is defined as follows: the Neighbor Type is set to the new CH_NEIGH value and the Link Type field is set to the UNSPEC_LINK value. The Neighbor Interface Address list is composed of one address, the clusterhead address of the node sending the Hello message. Note that even if the address of the clusterhead is advertised similarly than a neighbor address, the clusterhead may not be a neighbor of the node. When a "regular" OLSR node receives such an Hello message, it discards the clusterhead related part since it does not understand the link code but the other blocks of the message advertising the neighborhood status are processed as usual.

Upon reception of a Hello message with a clusterhead address, the clusterhead address must be saved. We choose to save it in the neighbor set of the sending node. Therefore, a new field, the N_clusterhead_address field, has been added to the neighbor tuple which was previously made of three fields. Therefore, a node records a set of "neighbor tuples" (N_neighbor_main_addr, N_status, N_willingness, N_clusterhead_address), describing neighbors. N_neighbor_main_addr is the main address of a neighbor, N_status specifies if the node is NOT_SYM or SYM. N_willingness is an integer between 0 and 7, and specifies the node's willingness to carry traffic on behalf of other nodes. N_clusterhead_address is the address of the clusterhead of the neighbor.

Upon receiving a Hello message, a node should update its Neighbor Set as follows:

- if the Originator Address is the N_neighbor_main_addr from a neighbor tuple included in the Neighbor Set: then, the neighbor tuple should be updated as follows:
  - N_willingness = willingness from the Hello message
  - N_clusterhead_address = Neighbor Interface Address associated with the Link Code where Neighbor Type = CH_NEIGH and Link Type = UNSPEC_LINK

6.2.3 TC messages

A regular version of the OLSR protocol is used within each cluster. To limit the propagation of the TC messages to the cluster area, TC messages are never forwarded by a node that does not belong to the same cluster as the originator of the TC messages. The MPR selection algorithm is performed without any consideration of the clusters for network consistency purposes. Therefore, each MPR sends periodic TC messages containing the list of its MPR selectors, i.e. the nodes which select it as MPR. When receiving a TC message, a node processes it following the algorithm described in RFC 3626 [28]. The forwarding decision is then based on the clusterhead of the sender, i.e. the node that has just forwarded the message (and not the originator of the message). The TC forwarding algorithm is roughly the same as the default forwarding algorithm described in [28] except the first step.

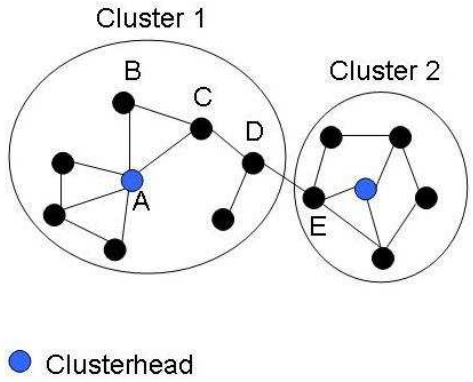

Figure 6.4 Illustration of the TC propagation

TC Forwarding Algorithm - Step One
If the sender interface address of the message is not detected to be in the symmetrical 1-hop neighborhood of the node, **or if the clusterhead address corresponding to the sender interface address is not the same as our clusterhead address**, the forwarding algorithm MUST silently stop here (and the message MUST NOT be forwarded).

There is no need for the originator of a TC message to add its clusterhead address in its TC messages. Indeed, a node is able to know which is the clusterhead of the node having forwarded the message thanks to the information contained in its Neighbor Set. Moreover, the clusterhead of the node that has forwarded the message (the sender) is necessarily the same as the clusterhead of the node that has previously forwarded the message to it, otherwise the message would not have been forwarded. Step by step, the clusterhead of the TC message originator is the same than the clusterhead of the sending node of a received TC message. Let us consider the example illustrated by the figure 6.4. Node B generates a TC message. Upon reception, node C looks in its Neighbor Set table for the address of the clusterhead of B. Node C finds that B has the same clusterhead than itself. Therefore, C can forward the TC message. Upon reception, D performs a look in its Neighbor Set for the clusterhead address of C. C and D have the same clusterhead. Therefore, D can forward the message. When receiving the message, E looks in its Neighbor Set table for the clusterhead address of the sender of the message which is node D. E and D does not have the same clusterheads since they do not belong to the same cluster. Therefore, E does not forward the TC message originated by B. B does not include its clusterhead address in its TC message and yet E is able to know that it does not have to forward the TC message anymore. The propagation of the TC messages is actually restricted to the cluster area.

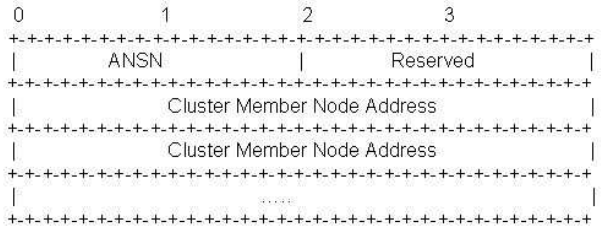

Figure 6.5 TC_Cluster message format

6.2.4 TC_Cluster message

From the previously described protocol, a node is able to compute a route to all the nodes in its cluster. Border nodes[2] are also able to compute routes to nodes belonging to neighbor clusters since they receive (but not forward) TC messages generated within these clusters.

For the routes to nodes belonging to different clusters, our approach is that each node should know the next hop toward the clusterhead the destination depends on. In the preceding example, if B wants to send data to E, B only needs to know the Next Hop node on the path to the clusterhead of E. Then, once the data packet arrives in the cluster of the destination node, the intermediate node knows the exact route to the destination. To achieve such behavior, cluster topology information must be sent over the network. OLSR Tree or C-OLSR approach is to reproduce the OLSR protocol at the cluster level to create some "cluster paths". The approach we follow is different. We define a new TC_Cluster message that is sent by clusterheads over the network using the MPR flooding algorithm (fig 6.6). This message does not contain the list of the MPR selectors of the clusterhead but rather the addresses of nodes belonging to its cluster. Since this message is flooded on the overall network, each node can maintain a node/cluster membership table and can therefore determine to which cluster a destination node belongs to. Nevertheless, knowing the clusterhead the destination node is related to is not enough to route a packet toward this destination node. Indeed, the path to the clusterhead or at least the next hop node on the path to the clusterhead is also needed. This next hop information is retrieved when receiving the TC_Cluster message. Indeed, when a node receives a TC_Cluster message, it registers as its next hop to the clusterhead sending the message the node that has just forwarded the message, assuming that this is the first time this message is received. Since the message is flooded over the network, a node may receive several copies of a TC_Cluster message. Nevertheless, the first copy received is the only one considered for the next hop information since it has necessarily taken the faster, less congested path. The other copies are discarded.

Figure 6.5 gives the format of the TC_Cluster message. The originator of a TC_Cluster message is the clusterhead, and the source address in the IP header is the candidate next-hop node toward the clusterhead. The number of hops to reach the clusterhead can also be computed through the TC_Cluster message thanks to the following formula: $TTL\_TC\_Cluster$ -$TTL\_of\_the\_received\_message$, where

[2] A border router is a node that is one hop away from a node that belongs to a different cluster i.e. a node that has a different clusterhead.

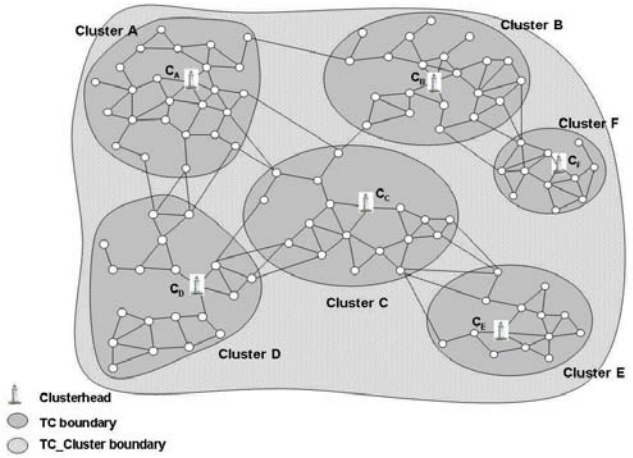

Figure 6.6 TC and TC_Cluster propagation boundaries

- TTL_TC_Cluster is a constant and is the TTL value the originator of a TC_Cluster message must set in the TTL field of the message

- TTL_of_the_received_message is the TTL value indicated in the TTL field of the received TC_Cluster message.

The TC_Cluster periodicity is lower than the TC periodicity assuming that the clustering protocol creates stable clusters.

6.2.5 Sending and forwarding data packets

When a node has a data packet to send:

- If it knows the destination from its routing table, it means that the destination node belongs to its cluster or that the destination node is a clusterhead. Therefore, it sends the packet to the next hop indicated in its routing table.

- If the destination is not in its routing table, it performs a look in the node/cluster membership table to know which cluster the node belongs to.

  - If the destination is not in the table, the packet is discarded.
  - If the destination is in the table, the node looks into its routing table for the next hop toward the clusterhead associated to the destination node address. The destination address of the data packet is not changed. When the next hop receives the data, the same process is performed.

Therefore, hop-by-hop, as the data packet gets closer to the destination, it is routed based on the clusterhead address of the destination node. When the data packet reaches a node belonging to the same cluster than the destination node, the packet is routed based on the destination node address.

6.3 Performance analysis: theoretical results and simulation

In this section we want to evaluate the overhead generated by our protocol. Since our objective is mainly to reduce the control overhead caused by the TC messages, we will only consider the TC message control overhead. For the analysis of the performance of our protocol, we are going to have an iterative approach. Indeed, we go on step by step and a step needs to be validated before moving on to the following one. The first step consists in evaluating roughly the overhead caused by the control messages through a theoretical analysis in order to evaluate if our solution presents better or at least similar performances than its counterparts. Then, once this theoretical analysis is validated, we implement the protocol in a simulator which have an ideal MAC layer and no mobility. Once these two steps are validated, we implement the protocol in a discrete event simulator such as OPNET in order to evaluate the performance of the protocol with a realistic MAC layer and with mobility. In this third step, the mobility is random and the applications profiles are ordinary. Then, in a last step, we evaluate the performance of the protocol with mobility and applications profiles that are more representative of the military context. In this chapter, we present the first two steps of this iterative approach. Firstly, we give theoretical analyses of the control overhead of our protocol and of the Fisheye OLSR protocol in order to verify that employing a clustering approach allows to improve or at least to reach a similar control overhead comparing to the Fisheye technique. Then, we compare our approach to the C-OLSR and the Fisheye OLSR protocols through a simulation study.

6.3.1 Theoretical analysis

6.3.1.1 Network model and parameters

Mobile Ad hoc Networks are often modeled thanks to a graph. To each terminal of the network corresponds a vertex of the graph and it exists a edge between vertex A and vertex B if the node corresponding to B can hear the packets sent by the node corresponding to A. If all nodes have the same emission power and if the radio system is ideal, then the propagation range is circular. Therefore, all nodes at a distance inferior to R from A can hear the signal sent by the node A. Such a network can be modeled by the Unit Disk Graph model [26].

In a Unit Disk Graph, there is an edge between two nodes A and B if and only if the Euclidean distance between A and B is at most 1. Unit disk graphs have proved to be useful in modeling various physical real world problems. One prominent application of unit disk graphs can be found in the field of wireless networking, where a Unit Disk Graph represents an idealized multi-hop radio network. Nodes are located in the Euclidean plane and are assumed to have identical (unit) transmission radii. They can communicate only if they are within mutual transmission range. Clearly, the Unit Disk Graph has become a standard when studying ad hoc networks. It is also obvious that the unit disk graph model is not fully realistic since it does not take into account interferences between simultaneous transmitters, or obstacles or even the fact that every node does not have the same power of emission. Nevertheless,

Unit Disk Graphs have well-known properties and provide enough information to achieve the objectives of the first step of our performance evaluation process. Therefore even if several other models have been proposed to study ad hoc networks ([71]), we choose to employ the Unit Disk Graph model. The neighborhood of a node A is made of all nodes contained in a circle of radius 1 with A, the center of the circle.

In the following and inspired from [2], the network is represented by a Poisson Point Process over the plan denoted S with intensity λ. Let N be the number of nodes in the network. N follows a Poisson law with intensity $\lambda * S$. λ represents the mean number of nodes per unity of surface. It follows that the density of the network $M = \lambda$, which means that on average each node has M neighbors or that on a unit disk centered on a node, there are on average M nodes. Therefore, the number of nodes in the K-hop neighborhood of a node is equal to the number of nodes in a disk a radius K which is on average K^2M. Moreover, the radius of the network is $\sqrt{N/M}$. Let M_R be the average number of MPRs selected by a node with a neighborhood size M. It has been shown in [60] and [2] that $M_R \leqslant (9\pi^2 M)^{1/3}$ and that $M_R \sim \beta M^{1/3}$ when $M \to \infty$ with $\beta \approx 5$.

The number of retransmissions of a TC message in the K-Hop neighborhood is equal to the number of nodes selected as MPR in the K-Hop neighborhood, which is on average equal to the number of nodes in the K-Hop neighborhood times the probability for a node to be selected as MPR by a neighbor node. Consequently, the number of retransmissions of a TC message in the K-Hop neighborhood of a node is on average :

$$M_R/M * K^2M = M_R K^2 \tag{6.1}$$

Then it follows that the number of nodes at exactly K hops of a node that may retransmit a TC message is on average :

$$M_R/M * (K^2 - (K-1)^2)M = M_R(K^2 - (K-1)^2). \tag{6.2}$$

6.3.1.2 Fisheye OLSR

In the Fisheye OLSR improvement [2], the period of the TC messages received from a node increases with the distance to the sending node. We can define a function F that gives the frequency of the TC messages based on the number of hops from the source, i.e. the TTL set in the messages. Let us consider the function $F : F(x) = \frac{4}{3+x^2}$ where x represents the TTL and $\frac{F(x)}{TC_{Interval}}$ is the frequency of the TC message for the TTL x. $TC_{Interval}$ is the period of the TC message. The overhead generated by a TC message sent by a node selected as MPR in bits/s is :

$$OH = \sum_{K=1}^{\sqrt{N/M}} F(K)(((K^2 - (K-1)^2))\frac{M_R}{TC_{Interval}} * TC_{size} \tag{6.3}$$

TC_{size} is the size of a TC message in bits. Finally it follows that the overhead in bits/s due to the TC messages in the Fisheye OLSR protocol is on average, since each network node can be elected as an MPR by one of its neighbor nodes:

$$(OH) * \text{number of node selected as MPR in the network} = (OH) * N \tag{6.4}$$

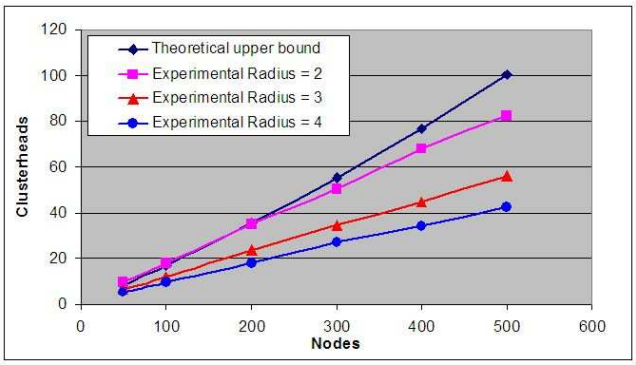

Figure 6.7 Mean number of clusterheads vs. number of nodes

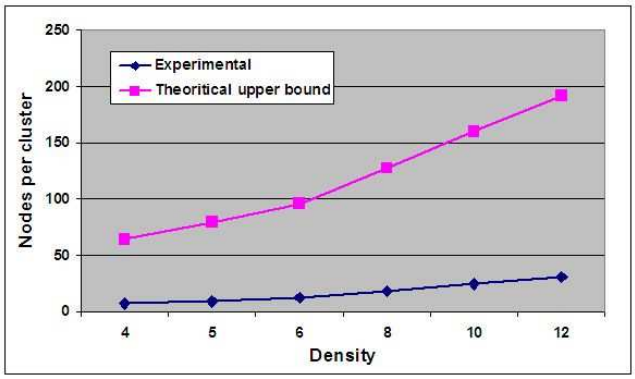

Figure 6.8 Upper bound and practical number of nodes per cluster vs. density

6.3.1.3 Our approach

Let $TC_{Interval}$ be the period of the TC message. The default value is 5 seconds. Let C be the mean number of clusters in the network. It has be shown in [33] that for the Max-Min heuristic [5], an upper bound of the mean number of clusters can be found :

$$\begin{aligned} &\mathbf{E}\left[\text{Clusterhead Number in S}\right] \\ &\leqslant \lambda \cdot \nu(S) \cdot \left(1 + \sum_{n=1}^{\infty} \frac{1}{n} \frac{E^n}{n!}\right) exp\left(-E\right) \end{aligned} \tag{6.5}$$

with $E = \lambda \pi R^2$, where R is the propagation range of a node and $\nu(S)$ is the Lebesgue measure of S. This upper bound is computed for a radius of 1. It is shown that for radius greater than 1, the set of clusterheads is included in the one computed for radius 1. Therefore, this upper bound becomes less and less accurate, as the radius increases as illustrated by figure 6.7, where the theoretical upper bound as well as several simulated mean numbers of clusterheads for various radius are represented.

In our approach, we have to distinguish the overhead generated by TC messages forwarding within each cluster from the overhead generated by the forwarding of the TC_Cluster messages. An upper bound of the mean number of nodes per cluster is equal to $r^2 M$, where r is the radius of the cluster. Fig 6.8 illustrates the discrepancy

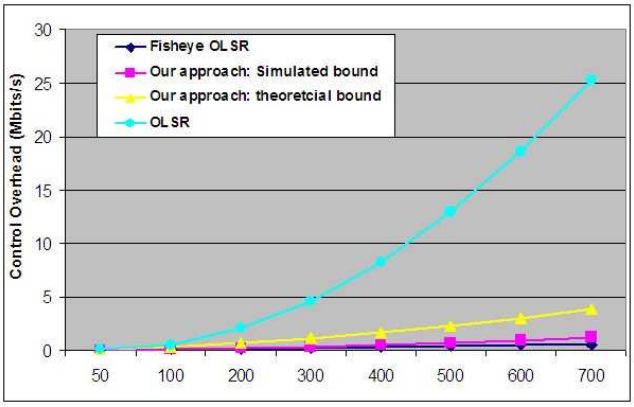

Figure 6.9 TC overhead comparison between Fisheye OLSR and our protocol versus the number of nodes

between the theoretical upper bound and the practical number of nodes obtained through simulation of the generalized max-min clustering algorithm [33]. The overhead due to the forwarding of a TC message sent by an MPR within a cluster in bits/s is thus bounded by the following upper bound:

$$B = (1 + r^2 M_R) * TC_{size}/TC_{Interval} \tag{6.6}$$

Moreover, the mean overhead generated by the forwarding of a TC_Cluster message sent by a clusterhead in bits/s is :

$$M_{OH} = (1 + NM_R/M) * TC\_Cluster_{size}/TC\_Cluster_{Interval} \tag{6.7}$$

Finally, the control overhead due to the TC and TC_Cluster message forwarding is bounded:

$$\mathbf{E}\,[\text{control message overhead}] \leq (B) * (N) + (M_{OH}) * C \tag{6.8}$$

6.3.1.4 Comparison of the theoretical bounds of the control overhead

In this section, we compare the theoretical overhead of the Fisheye OLSR and our proposal based on the expressions given in the previous sections. We also implement the generalized max-min clustering algorithm in order to obtain simulated values for the mean number of clusters and for the mean number of nodes per cluster. Indeed, figures 6.7 and 6.8 show that there can be important differences between the theoretical upper bounds and the simulated values for these two parameters. Therefore, we trace two curves, the "Our approach: theoretical bound" where the values of the mean number of clusters and the mean number of nodes per cluster are replaced by their theoretical values and "Our approach: simulated bound" where the values of the two parameters are replaced by their simulated values. One should note that in the following results, the overhead due to the clustering algorithm has been added to the TC and TC_Cluster control overhead for our solution.

As illustrated by figure 6.9, when the number of nodes increases, the overhead due to the TC messages increases slowly with both the Fisheye OLSR solution and our OLSR clustering algorithm. For these experiments, the density is set to 6 neighbors per node and the radius of a cluster is set to 3. One should note that the overhead

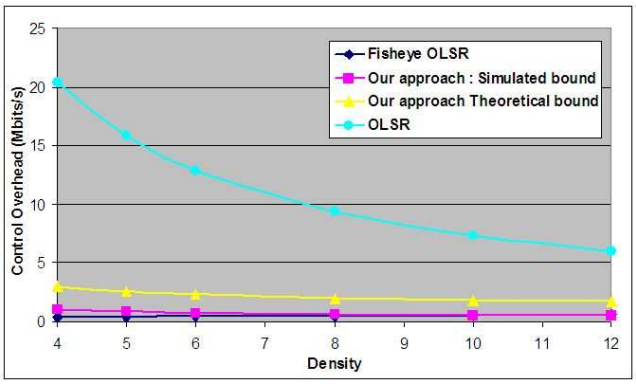

Figure 6.10 TC overhead comparison between Fisheye OLSR and our protocol versus the density

of our approach obtained through simulation is less than the theoretical overhead which is based on an upper bound of the mean number of clusters and on the mean number of nodes per cluster. This bound is over-estimated and is therefore higher than the theoretical overhead of the Fisheye OLSR solution. Figure 6.10 presents the overhead of the TC messages in a network of 500 nodes as the density increases. It shows that the performance of all approaches improves as the density increase. This result is due to the fact that when the density increases, the MPR-based forwarding algorithm is more efficient. Indeed, fewer retransmissions are needed to send TC messages to the network. However, Fisheye OLSR seems to slightly outperform our approach. Nevertheless, even with the simulated values for the number of clusterhead and the number of nodes per cluster, the results presented for our approach are still upper bounds. In the second step of this performance evaluation, we implement the protocols in a simulator with an ideal MAC layer and no mobility to verify these results.

6.3.2 Performance evaluation based on simulation

In this section, we compare the overhead of different solutions to improve the scalability of OLSR. We consider the Fisheye OLSR solution and the C-OLSR solution that we compare to our approach. These three protocols have been implemented thanks to the Scilab 4.1.2 [115] simulation tool. For the clustering, we implement the generalized max-min clustering algorithm. Since we are mainly interested in the control overhead, we compare the control overhead caused by either the TC messages or their substitutes in each of these protocols:

- the TC messages for the Fisheye OLSR protocol
- the TC messages forwarded within each cluster, the C-Hello messages, the C-TC messages and the control message overhead due to the clustering protocol for C-OLSR
- the TC messages forwarded within each cluster, the TC_Cluster messages, and the control message overhead due to the clustering protocol for our protocol.

Figure 6.11 presents a comparison between the theoretical upper bound and the simulated values of the overhead of the Fisheye OLSR and our solution. For these

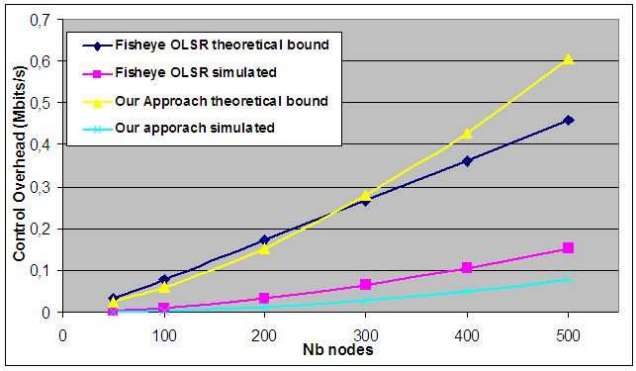

Figure 6.11 Comparison of the theoretical values and the simulated of the overhead

experiments, the density is set to 8 neighbors per node and the radius of the cluster is set to 3. These results show that the bounds are rather far from the practical values. Moreover, if the theoretical bound of our approach is higher than the theoretical bound of Fisheye OLSR for large networks, the simulation shows that even for large networks, our approach performs better than Fisheye OLSR.

Figure 6.12 presents a comparison of the control overhead of Fisheye OLSR, C-OLSR and our solution as a function of the number of nodes (the density is set to 8 neighbors per nodes and the radius of the clusters is set to 3 hops at a maximum). The results prove that employing a clustering algorithm does not necessarily allow to improve the scalability of OLSR compared to the Fisheye solution. Moreover, we show that our solution presents better scalability compliance than the C-OLSR solution where OLSR is applied on top of the cluster topology. The difference between the overhead of C-OLSR and our approach results only from the inter-cluster communications approach since both the clustering and the intra-cluster TC forwarding are similar in these two solutions. Therefore, the results prove that a solution that does not re-use the OLSR algorithm on top of the cluster topology presents better efficiency in term of control overhead than applying an adapted version of OLSR on the cluster topology.

6.4 Conclusion

This chapter presents a scalable routing protocol for tactical Mobile Ad hoc Networks that is based on and improves the well known proactive unicast routing protocol OLSR to make it scalable. The protocol assumes that nodes are gathered into clusters thanks to a clustering algorithm. The regular OLSR protocol is applied within the clusters for intra-cluster communications. Protocols that take a similar approach such as C-OLSR or OLSR Tree choose to re-use an adapted version of OLSR on top of the cluster topology. We choose to follow a different approach for the inter-cluster communications where we do not re-use a version of OLSR on the cluster topology since we expect that such a solution may generate an important overhead. Therefore, a new message type is defined for the inter-cluster routing. This new message, called TC_Cluster, is sent by each clusterhead and contains the list of the nodes belonging to their cluster. TC_Cluster messages are broadcasted

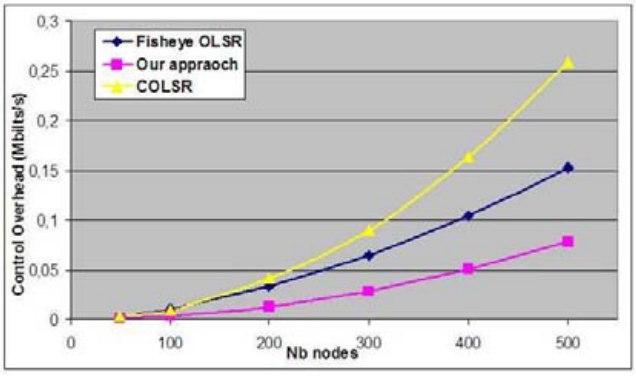

Figure 6.12 Comparison of the control overhead of Fisheye OLSR, C-OLSR and our approach

over the entire network thanks to the optimized MPR flooding.

Theoretical analyses of the control overhead of our protocol show that our approach allow to reduce the control overhead as the number of nodes in the network increases compared to the OLSR protocol. It also show that employing a clustering technique allow to obtain similar results than the Fisheye OLSR protocol. The simulation analysis shows that depending on the protocol employed for the inter-cluster unicast routing the results can be different. Indeed, as we expect, our solution for the inter-cluster communications allows to significantly reduce the control overhead compared to the C-OLSR solution. Moreover, our solution obtain lower control overhead than the Fisheye OLSR solution whereas the C-OLSR protocol presents poorer performance. The next step of our performance evaluation will consist in implementing the protocols in a discrete event simulator to verify that these results are still valid with a realistic MAC layer and with mobility.

Chapter 7

Conclusion and Perspectives

7.1 Conclusion

The Transformation of the military networks that is currently occurring with the advent of the Network Centric Warfare adopts the concept of "Mobile Ad Hoc Network" (MANET) as a central component of the tactical network environment. Military networks are typically transportable networks made of multiple components among which the tactical network which refers to the nodes responsible for the operations on-the-field. In the future, the tactical network is expected to be made of nodes that are mobile, self-managing, self-configuring without relaying on any infrastructure. Thereby, the concept of Mobile Ad hoc Networks appears as the ideal candidate solution for supporting the fully mobile and dynamic tactical communication networks.

Since the 1980's, the Mobile Ad Hoc Networks have known a substantial attention. MANETs are often considered as "wireless access network" solutions to connect mobile users to a fixed infrastructure. Their use in the military environment is different from the commercial employment. Whereas in the commercial use of mobile ad hoc network the wireless network is seen as an extension of a wired IP network i.e. operating as a stub, its use within the tactical network structure places it as a transit network carrying traffic entering and then leaving the network (and not sinked or generated by MANET nodes). The first chapter of this manuscript presents the tactical network architecture and the role of the tactical MANET as a transit network within the architecture. The Tactical Internet is made of a variety of heterogeneous networks such as LANs, satellite networks, commercial networks that are interconnected through the Tactical Communications Nodes. These Tactical Communications Nodes integrate radio equipments to form together a highly dynamic radio network, the tactical MANET. Thereby, the tactical MANET is a transit network that must inter-operate different networks (LANs, commercial Internet...). The context of employment of the tactical MANET engenders challenges to solve as well as the ones inherent to this type of network. We can mention among others the scalability (tactical networks can be made of several hundreds of nodes), the importance of multicast communications and the interoperability with wired networks. Through this thesis, we endeavours to define how multicast communications can be settled between actors that are spread among different types of networks interconnected thanks to the MANET network.

In the second chapter, we study the protocol architecture that must be implemented to provide a multicast service in the tactical network. We particularly pay attention to the way the multicast service must be implemented within the tactical MANET knowing that the tactical MANET is interconnected with different types of networks implementing different multicast protocols. Three solutions depending on the degree of adaptation of the multicast service to the MANET environment, going from no multicast protocol to a MANET-specific multicast protocol, with a wired multicast protocol as an intermediate solution, have been considered. We conclude that the MANET-specific solution is the best approach to provide efficient multicast service in the tactical network environment. This choice underlines the need for a dedicated and specific multicast routing protocol within the tactical MANET and also raises interoperability issues meaning that interconnection or proxying solutions should be implemented at the interface with the wired IP networks. Since scalability is a major constraint in this environment, strategies to provide scalability in

MANET are studied and compared. We come to the conclusion that the clustering approach, which gathers nodes into groups, presents the most promising characteristics to achieve the scalability objective of the tactical MANET. Therefore, we distinguish two levels of multicast communications: the intra-cluster multicast communications when the multicast members are located within the same cluster and the inter-cluster multicast communications when the multicast members belong to different clusters. It underlines the need for an intra-cluster multicast routing protocol that is responsible for the multicast flows within each cluster and an inter-cluster multicast routing protocol that is responsible for the multicast flows from cluster to cluster.

The third chapter focuses on the intra-cluster multicast routing issue. We present a new vision of the state-of-the-art of multicast routing protocol for MANET that takes the design objective (robustness, efficiency ...) as a criteria rather than a characteristic of the protocol such as the topology of the structure or the route acquisition scheme. This review underlines the lack of protocols that can provide high delivery guarantees with low overhead i.e. that is robust AND efficient. Consequently, we propose a new protocol called STAMP for Shared-Tree Ad hoc Multicast Protocol as an alternative to the existing flat multicast routing protocols by combining into one protocol the robustness and the efficiency compliance. The efficiency requirement is met thanks to a shared-tree structure maintained through a hard-state approach and where the initiative of the shared tree construction is given to the group members. For the robustness, STAMP takes advantages of the broadcast capacity of the medium to introduce redundancy without increasing the data overhead. Thanks to a performance evaluation study, in which we compare STAMP to the well-known mesh-based multicast routing protocol ODMRP, we demonstrate that the goal of robustness and efficiency are achieved. In scenarios where the tree-based protocols are known to fail in term of packet delivery ratio, when the mobility increases for example, STAMP achieves high packet delivery ratios comparable to the mesh-based ones. All the more that this high delivery guarantees are achieved with a high efficiency and not at the expense of a high data and control overhead as for mesh-based protocols.

The fourth chapter presents the ScAlable structure-Free Inter-cluster Multicast Routing Protocol (SAFIR) as a solution to the inter-cluster multicast routing issue. SAFIR is responsible for handling the inter-cluster multicast communications and assumes that an intra-cluster multicast routing protocol such as STAMP is applied within each cluster. Therefore, the aim of the protocol is to define how a multicast datagram for a group can be forwarded from cluster to cluster until reaching the clusters where the multicast members for the multicast group are. Our protocol is optimized in term of efficiency (control and data overhead) since it benefits from the other services (unicast, clustering ...) control messages to send the information needed for its operating. Moreover, unlike other existing protocols, SAFIR does not rely on any join/reply/leave messages to construct a multicast delivery structure on the cluster topology. In this chapter, we also present the way the two levels of multicast protocols may interact to achieve seamless end-to-end communications within the tactical MANET. The performance evaluation study we fulfill on SAFIR confirms that SAFIR meets the scalability constraint and that the association of SAFIR and STAMP presents interesting results compared to the association of SAFIR and

ODMRP or to ODMRP on a flat network.

In the fifth chapter, we examine the interoperability issues between the multicast routing protocols defined for the tactical MANET, i.e. STAMP and SAFIR and the multicast protocols that may be deployed in external IP networks and local LANs. We study the repercussions on the multicast service of the repartition of the different multicast actors among the different types of the wired IP networks. Consequently, we identify several points that needs to be addressed in order that the different protocols can interact and provide end-to-end seamless multicast capability through the tactical MANET. Solutions to these different issues are proposed. At the end of this chapter, the protocol architecture of the multicast service that must be deployed within the tactical MANET to provide end-to-end seamless multicast service within the tactical network is fully defined.

Even if we focus mainly on multicast communications through this thesis, an efficient unicast routing service is also needed within the tactical MANET. After a review of the literature concerning unicast routing in MANET, it appears that the subject of scalable unicast routing relying on a clustered network is only partially studied. This last chapter presents a scalable routing protocol for tactical Mobile Ad hoc Networks that is based on and improves the well-known proactive unicast routing protocol OLSR to make it scalable. The protocol we define is different from the other protocols that share the same objectives since we do not choose to apply a version of OLSR on the cluster topology. Indeed, we expect that such a solution generates an important overhead. Therefore, a new message sent by the clusterheads and containing the list of the nodes belonging to their cluster is defined for the inter-cluster routing. Theoretical analyses of the control overhead of our protocol compared to the other solutions not based on clustering show that our approach significantly reduces the control overhead as the number of nodes in the network increases. Moreover, a simulation study where we compare our solution to other protocols that propose to enhance OLSR and that are cluster-based, proves that our innovating solution for the inter-cluster communications allows to reduce the control overhead significantly.

7.2 Perspectives

As possible directions for future works concerning the topics studied in this thesis, we outline in this part some ideas:

- The data delivery of STAMP is performed on the tree like in a mesh. An enhancement of this solution can be to adapt the data delivery of STAMP depending on the network conditions such as the mobility or the traffic load. Thereby, the data delivery may switch from a mesh-based forwarding when the network conditions are disadvantageous (important mobility...) to a tree-based forwarding when the network conditions allow it. This would allow us to decrease the data overhead even more. Moreover, if the MAC layer is adapted, it would allow us to optimize the bandwidth use.

- As far as performance evaluation is concerned, several works can be done. Firstly, a simulation model of the global network architecture can be developed to evaluate the performance of the different solutions proposed for the

interconnection. Then, to go further in the performance evaluation of the cluster-based OLSR enhanced protocol, a simulation model of the protocol can be developed to evaluate end-to-end performance like the packet delivery ratio or the end-to-end delay. Finally, a simulation model that integrates all the proposed protocols (STAMP, SAFIR, the cluster-based OLSR enhancement and the solutions of interoperability) can also be considered. At this time, we have already integrated STAMP and SAFIR in the modeling scheme.

- The performance evaluation work we propose in this thesis aims at evaluating the protocols in a general framework, with random placement and random mobility, or with the 802.11e model as the MAC layer for example. This is the first phase of the performance evaluation. The next phase would be to evaluate the protocols in a more relevant environment. For instance, the MAC layer can be replaced by a MAC layer that is more representative to the target system on which the protocols have to be deployed. Similarly, the scenario chosen for the evaluation can be more representative to operational scenarios as far as the mobility, the number of nodes, the placement, the traffic load, the multicast members repartition are concerned. Finally, the protocols can also be evaluated with real application traffic rather than CBR traffic source. For instance, the System-In-The-Loop module of OPNET can be used to inject real application traffic such as a video stream into the simulation.

- Finally, the ultimate step would be to implement the protocols proposed in the final product to evaluate the performance of the protocols in real deployment scenarios. Such a step is supposed to be performed for STAMP the coming year.

Additionally to these perspectives that refer to improvement of the work presented through the thesis, we can consider general directions for further research. Now that an efficient and robust multicast service has been defined, it would be meaningful to address the other requirements that a multicast routing protocol must provide for use in a tactical environment, i.e. the QoS, reliability and security. Indeed, the multicast routing protocol architecture defined in this thesis thanks to SAFIR and STAMP can be enriched by additional mechanisms to meet the list of requirements presented in chapter 2. The next step would be to incorporate QoS routing possibilities, security features such as a group key management scheme and reliability protocols. For each of these services, interoperability studies need to be realized so that an end-to-end seamless multicast service can be provided to the user. Finally, IPv6 is expected to be the future of the Internet Protocol that is today the IPv4. IPv6 offers solutions to the limitations of IPv4 which was not designed for the type of Internet-work that the Internet has become. Therefore, in the context of multicasting in MANET networks, some researches need to be performed to evaluate the impact of the migration to IPv6. This work includes the evaluation of the STAMP and SAFIR protocols in order to identify the changes that need to be done so that those protocols can be integrated in an IPv6 environment.

Annexes

.1 Annexe 1: Description of some Internet multicast-related protocols

This annexe provides brief descriptions of some of the protocols involved in the Internet Multicast model, beginning by the IGMP protocol. Then, three intra-domain and two inter-domain multicast routing protocols are presented.

.1.1 Internet Group Management Protocol (IGMP)

IGMP is a protocol implemented within the IP stack of a host or a router. It is used between a host and its local router. The purpose of IGMP is to manage dynamically the multicast group membership on a local LAN i.e. between a router and its local hosts. The first version of the protocol defines the basic operation of the protocol (IGMPv1) [34]. Then, two other versions (IGMPv2 [42] and IGMPv3 [15]) have been defined to provide additional functionalities.

In its basic functioning (IGMPv1), two messages are used.

- General query message: a router sends periodically a general query message to ask to its directly attached hosts what their multicast group memberships are;
- Membership report message: when receiving a general query message or each time its group membership changes, a host sends a membership report message to its local router to inform it about its group membership. In order to avoid that the replies from each host arrive at the same time, each host waits during a random period of time (from 1 to 10 seconds) before replying. If a host replies for a group G, all the reports for the same group are discarded (this assumes that a host hears reports sent by the other hosts).

The router keeps a list of the multicast group membership for each attached LAN where the "multicast group membership" is the list of the multicast groups for which there is at least one member in the LAN. The router does not keep the list of addresses of all the members. To leave a group, a host has only to stop responding to the requests for this group. This may imply a more or less important delay from the moment when the host wants to leave the group to the moment when it effectively quits the group. Figure 1 provides an illustrative example with two hosts

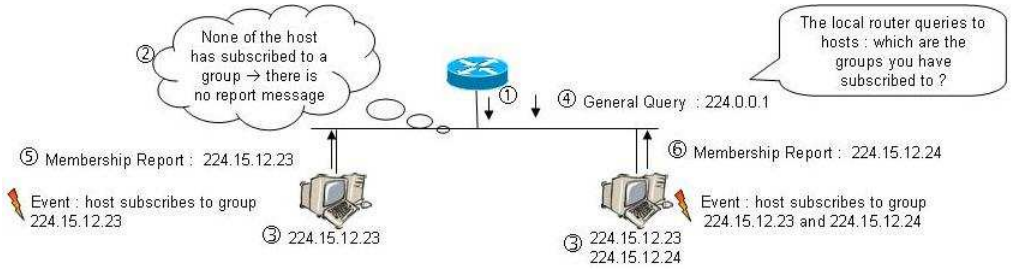

Figure 1 An illustrative example of the IGMP operation

linked to their local router. At the beginning, the hosts are not member of any group. First, the router sends a general query to the hosts to know whose groups

they want to belong to the multicast session (1). As none of the two hosts has subscribed to any group, there is no report message (2). Then, an application event occurs and the hosts subscribe to groups: the one on the left wants to be member of group 224.15.12.23 and the one on the right wants to be member of both groups 224.15.12.23 and 224.15.12.24 (3). After a predefined period of time, the router sends a general query (4). The address 244.0.0.1 defines the group composed of all hosts. The host on the left is the first to respond (because it has a shorter timer) and sends a membership report message with the multicast group address 224.15.12.23 (5). Then, the host on the right sends its membership report with the multicast group address 224.15.12.24 (6). Note that this host does not report the address 224.15.12.23 whereas it is a member of this group. Indeed, the host on the right has heard the report message of the host on the left. Consequently, it knows that the group 224.15.12.23 has already been notified to the router.

IGMPv2 [42] has basically the same operating than the v1. It allows group membership terminations to be quickly reported to the router and to the routing protocol which could be important for broadband multicast groups or for subnets with highly volatile group memberships. In the preceding version, a router could detect that a host leaves the group only by the lack of response to a query message. The version 2 defines a new message that will enable hosts to leave groups quickly. A leave message is sent by a host to his local router to inform it about its group leaving. Moreover, the router is given the ability to send a query for a specific group with the group specific query message.

IGMPv3 [15] adds the support for "source filtering". It is the ability for a system to report interest in receiving packets sent to a particular multicast address only from specific source addresses or from all but specific source addresses. The message format has been changed in order to provide a host the ability to precise the sources it wants to include or exclude.

.1.2 Examples of Intra-Domain Multicast Routing Protocols

.1.2.1 Distance Vector Multicast Routing Protocol (DVMRP)

DVMRP is an intra-domain distance-vector multicast routing protocol. It was the first protocol developed to support multicast routing and it is widely used in the Internet MBone [39]. The original specification was derived from the Routing Information Protocol (RIP) and implemented the Truncated Reverse Path Broadcasting (TRPB) algorithm. In its latest version, DVMRP employs the Reverse Path Multicasting (RPM) algorithm. DVMRP uses the Dense Mode approach and creates trees that are rooted at the source, called source-trees. DVMRP does not need any underlying unicast routing protocol since it computes itself its needed routing information.

According to the RPM algorithm, the first packet for any (source, group) pair is flooded to the entire network. The interfaces by which the packet is forwarded are stored in the outgoing interface list. Therefore, this first packet is delivered to all leaf routers, which send prune messages back to the source if there are no group members on their directly attached subnetworks. The prune messages result in the removal of the useless branches of the tree. When a router receives a prune message, it removes the interface by which the message has been received from the outgoing

interface list. The procedure creates a source specific shortest path tree where all leaves are routers that have group members on their behalf. Note that a leaf router is a router that does not have any outgoing interface for the multicast protocol. The initial procedure is repeated periodically.

Furthermore, DVMRP implements "graft messages" that allow to rapidly (without waiting for the next flooding) graft back a branch that has previously been pruned. If a router that has previously sent a prune message for a pair (source, group) discovers new members for this pair, it sends a graft message to the next hop back to the source. When a router receives a graft message, it cancels the previously received prune message and sends a "graft ack" to the router that has sent it.

When a router receives a multicast data packet for a group, if it belongs to the tree constructed for this group (if its outgoing interface list is not empty), it forwards a copy of the data on all interfaces of the outgoing interface list. DVMRP implements the poison reverse technique to reduce overhead i.e. before forwarding an incoming packet to a given interface, DVMRP makes sure that this outgoing interface leads to a router that will recognize the sending router as a router on the shortest path between it and the source.

.1.2.2 Protocol Independent Multicast (PIM)

PIM is the most recent protocol of all the multicast routing protocols. Contrary to the other multicast routing protocol, PIM supports both the dense and the sparse routing mode. Indeed, two versions of the protocol have been defined: PIM-Dense Mode (PIM-DM) and PIM-Sparse Mode (PIM-SM). As indicated by their name, PIM-DM is adapted for areas where the bandwidth is plentiful and the members are densely present whereas, PIM-SM is adapted for areas where the bandwidth is scarce and the members are sparsely dispersed in the network. The name PIM comes from the fact that the protocol is totally independent from the unicast protocol it is used with. PIM is able to operate whatever the unicast routing protocol is.

PIM-Dense Mode (DM) PIM-DM is similar to DVMRP in the sense that it uses the RPM algorithm with the two flood and prune phases. Nevertheless, there exist several differences between these two protocols described hereafter.

- PIM-DM relies on an existing unicast routing protocol to adapt to topology changes, but it is independent of the mechanism of this protocol. PIM-DM is thus simpler than DVMRP in the sense that it does not construct unicast routing tables. Remember that DVMRP contains an integrated routing protocol that makes use of its own RIP-like exchanges to compute the required unicast routing information;

- When receiving a packet on an interface, PIM-DM forwards the message on all of its interfaces until it receives a prune message whereas DVMRP computes the child interfaces for a pair (source, group) thanks to the poison reverse algorithm. With this last point, we can see two advantages of PIM regarding DVMRP: simplicity and saving resources from the routing tables.

After the flooding of the first packet, the leaf routers that do not have any members in their subnetworks send a prune message on the incoming interface. The prune

states have a finite lifetime during which a router that has pruned itself may graft to the tree thanks to a graft message that propagates from router to router toward the source. PIM-DM is a data oriented protocol, i.e. it is the emission of data that initiates the construction of the tree. No control information is exchanged before the first source starts sending data. PIM-DM is well suited for environments where the members are dense and the bandwidth is plentiful because in such conditions, the periodical flooding of a data message is not very penalizing compared to a network in which only few routers are interested in receiving the data and the bandwidth is scarce.

PIM - Sparse Mode (SM) PIM-SM has been developed as an alternative to dense mode multicast routing protocols in case of a sparse distribution of the group members. This protocol uses the Core Based Tree (CBT) forwarding algorithm where the core is called the rendezvous point" (RP), i.e. a meeting point where sources meet receivers or vice versa. A rendezvous point is elected in the network. When a router member wants to join the group, it sends a join message to the rendezvous point. Contrary to PIM-DM, this protocol is group-oriented (and not data oriented) in the sense that the initiative to begin the construction of the tree is given to the members. As long as members are present the structure must be maintained. Before any data exchange, the tree must be set up.

The PIM-SM join mechanism is the following.

- A receiver joins a group by sending a join message toward the RP, i.e. to the Next Hop router on the path to reach the RP. This information is given by the routing table constructed thanks to the unicast routing protocol;
- The join message is processed by all routers on the path between the receiver and the RP, which save the status information for the group. Therefore, a new branch of the shared tree is constructed.

When a source wants to send packet to a group:

- The source first encapsulates the multicast data in a unicast packet directed to the RP. A source only has to know the address of the RP and not the addresses of the group members. This phase is called the registration;
- Upon reception of a multicast packet, the RP will de-encapsulate it and forward it on all of its interfaces that belong to the distribution tree. These interfaces are the ones registered through the join mechanism.

PIM-SM offers the possibility for a receiver to switch to a source-rooted tree if the data rate of the source is over a certain threshold.

- The router sends a join message toward the source and a prune message toward the RP;
- Routers that are closer to the leaves of the RP multicast tree will also automatically switch to the source rooted tree route;
- The source will keep on sending a copy of its packet toward the RP considering it might be members of the group that are still receiving packet via the RP rooted tree.

Finally, PIM-SM uses semi-soft state, i.e. a state has to be refreshed by a join message periodically. If no joins are received during a time-out period the state entry is deleted.

.1.2.3 Multicast Open Shortest Path First (MOSPF)

MOSPF is the multicast extension of the OSPF protocol. Each OSPF router maintains a database of link state information which describes the network topology. This link state database is constructed thanks to the exchange of Link State Advertisement (LSA) messages. MOSPF extends OSPF by adding a new type of LSA message called the group membership LSA. These messages are flooded over the network so that each node in the network is aware of the network membership. Therefore, when a router receives a multicast data message, it is able to compute the shortest path tree rooted at the source of the packet and to forward the data packet accordingly.

.1.3 Examples of Inter-Domain Multicast Routing Protocols

.1.3.1 Border Gateway Multicast Protocol (BGMP)

BGMP is an inter-domain multicast routing protocol implemented by border routers that is able to inter-operate with any intra-domain multicast routing protocol. It constructs a bidirectional shared-based tree with other border routers. In BGMP, the root of the shared tree is an entire AS rather than a single router. The root AS for a particular multicast group address is the AS that has claimed the multicast address by employing a global multicast address allocation protocol such as the Multicast Address Set Claim (MASC) protocol.

BGMP runs on border routers and supposes that the border routers also run an intra-domain multicast routing protocol. It is thus up to the intra domain multicast routing protocol on the border router to gather multicast membership information in the domain and to give this information to the BGMP process. Such information triggers border routers to set up TCP connection between them and to exchange BGMP messages. Thus, when a multicast group membership changes, border routers send incremental join/prune message to one another. BGMP uses unicast routes advertised by the Border Gateway Protocol (BGP) to set up the TCP connection. Therefore, implementing BGMP implies the implementation of BGP as well as MASC.

.1.3.2 Multicast Source Discovery Protocol (MSDP)

The MSDP describes a solution to connect intra domain shared-trees without the need to construct an inter-domain multicast shared tree. MSDP was initially designed to operate with PIM-SM but it is also applicable to other shared-tree protocols (such as CBT) and to protocols that keep active source information at the border routers. MSDP is based on a different paradigm than protocols that construct an inter-domain tree between domains and then inter-operate with the intra domain multicast routing protocols to make sure the connection is maintained at the border routers. MSDP proposes that the presence of sources on other domains is known by the intra domain trees.

In MSDP operating, each RP of a domain maintains a MSDP peering session with the RPs of other domains through TCP sessions. The RPs that have active sources in their domains send Source-Active (SA) messages to their MSDP peers. These SA messages contain the IP address of the source, the multicast group address and the IP address of the RP that has sent the SA message. When receiving a SA message, a MSDP peer forwards it to its other MSDP peers. Moreover, it checks if it has multicast receiver for this group on its domain. If so, it triggers a join message to the source node on the MSDP peer's domain. This set up a branch of a source-tree to the domain.

Bibliography

[1] A. Adams, J. Nicholas, and W. Siadak, "Protocol Independent Multicast - Dense Mode (PIM-DM): Protocol Specification (Revised)," RFC 3973, January 2005.

[2] C. Adjih, E. Baccelli, T. Clausen, P. Jacquet, and G. Rodolakis, "Fish eye OLSR scaling properties," *Journal of communication and networks (JCN), Special Issue on Mobile Ad Hoc Wireless Networks*, vol. 6, no. 4, pp. 343–351, 2004.

[3] K. Al Agha, G. Pujolle, and G. Vivier, *Réseaux de mobiles et réseau sans fil GSM, GPRS, UMTS, 802.11, Bluetooth, BLR, DVB, IP Mobile*, Eyrolles, Ed. Eyrolles, October 2001.

[4] M. Almeida, R. Sarro, J. P. Barraca, S. Sargento, and R. L. Aguiar, "Experimental Evaluation of the Usage of Ad Hoc Networks as Stubs for Multiservice Networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2007, no. 1, pp. 39–52, 2007.

[5] A. Amis, R. Prakash, T. Vuong, and D. Huynh, "Max-min d-cluster formation in wireless ad hoc networks," in *IEEE INFOCOM*, vol. 11, Tel Aviv', Israel, March 2000, pp. 32–41.

[6] B. An and S. Papavassiliou, "A mobility-based hybrid multicast routing in mobile ad-hoc wireless networks," in *IEEE Military Communications Conf. (MILCOM2001)*, vol. 1, october 2001, pp. 316–320.

[7] S. Armstrong, A. Freier, and K. Marzullo, "Multicast Transport Protocol," RFC 1301, February 1992.

[8] I. D. Aron and S. K. S. Gupta, "On the scalability of on-demand routing protocols for mobile ad hoc networks: an analytical study," *Interconnection Networks*, vol. 2, no. 1, pp. 5–29, 2001.

[9] E. Baccelli, "OLSR Scaling with Hierarchical Routing and Dynamic Tree Clustering," in *IASTED International Conference on Networks and Communication Systems (NCS)*, Chiang Mai, Thailand, March 2006.

[10] A. Ballardie, "Core Based Trees (CBT version 2) Multicast Routing, Protocol Specification," RFC 2189, September 1997.

[11] T. Ballardie, R. Perlman, C.-Y. Lee, and J. Crowcroft, "Simple scalable internet multicast," Univ. College London, UCL Research Note RN/99/21, April 1999.

[12] E. Bommaiah, A. McAuley, R. Talpade, and M. Liu, "AMRoute: Adhoc multicast routing protocol," Internet Draft, August 1998.

[13] J. L. Burbank, P. F. Chimento, B. K. Haberman, and W. T. Kasch, "Key Challenges of Military Tactical Networking and the Elusive Promise of MANET Technology," *IEEE Communications Magazine*, vol. 44, pp. 39–45, November 2006.

[14] S. Cai, L. Wang, and X. Yang, "An Ad Hoc Multicast Protocol Based on Passive Data Acknowledgement," *Computer Communications*, vol. 27, no. 18, pp. 1812–1824, December 2004.

[15] B. Cain, S. Deering, I. Kouvelas, and A. Thyagarajan, "Internet Group Management Protocol version 3," RFC 3376, October 2002.

[16] T. Camp, J. Boleng, and V. Davies, "A Survey of Mobility Models for Ad Hoc Network Research," *Wireless Communications & Mobile Computing (WCMC) Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications*, vol. 2, pp. 483–502, 2002.

[17] A. K. Cebrowski and J. J. Garska, "Network-Centric Warfare: Its origin and future," *Proceedings magazine, United State Naval Institute*, January 1998.

[18] R. Chandra, V. Ramasubramanian, and K. P. Birman, "Anonymous Gossip: Improving Multicast Reliability in Mobile Ad Hoc Networks," in *21st International Conference on Distributed Computing Systems (ICDCS'01)*, Mesa, AZ, April 2001, pp. 275–283.

[19] M. Chatterjee, S. K. Das, and D. Turgut, "WCA: A weight-based distributed clustering algorithm for mobile ad hoc networks," *Journal of Cluster Computing (Special Issue on Mobile Ad hoc Networks)*, vol. 5, no. 2, pp. 193–204, 2002.

[20] K. Chen and K. Nahrstedt, "Effective location-guided tree construction algorithms for small group multicast in MANET," in *IEEE Infocom*, 2002, pp. 1180–89.

[21] T.-S. Chen, Y.-S. Chen, and H.-W. Tsai, "A Hierarchy-based Multicast Protocol for Wireless Mobile Ad-hoc Networks," in *Ninth IEEE International Conference on Networks*, October 2001, pp. 248–253.

[22] C. Chiang, H. Wu, W. Liu, and M. Gerla, "Routing in clustered multihop, mobile wireless networks with fading channel," in *SICON'97*, April 1997, pp. 197–211.

[23] C.-C. Chiang, M. Gerla, and L. Zhang, "Shared Tree Wireless Network Multicast," in *6th International Conference on Computer Communications and Networks*, Las Vegas, NV, USA, September 1997, pp. 28–33.

[24] C. Chiang, M. Gerla, and L. Xhang, "Adaptive Shared Tree Multicast in Mobile Wireless Netkworks," in *IEEE GLOBECOM'98*, vol. 3, 1998, pp. 1817–1822.

[25] C. Chiang, M. Gerla, and L. Zhang, "Forwarding Group Multicast Protocol (FGMP) for multihop mobile wireless networks," *Cluster Computing*, vol. 1, no. 2, pp. 187–196, 1998.

[26] C. C. Clark, B.N. and D. Johnson, "Unit Disk Graphs Discrete Mathematics," vol. 86, no. 1-3, pp. 165–177, 1990.

[27] T. Clausen, "Combining Temporal and Spatial Partial Topology for MANET routing - Merging OLSR and FSR," in *IEEE WPMC'03*, Yokosuka, Japan, 2003.

[28] T. Clausen and P. Jacquet, "Optimized Link State Routing Protocol (OLSR)," RFC 2636, October 2003.

[29] Commission de défense, "Les opérations réseau-centrées : les capacités européennes," Rapport l'Assemblée interparlementaire européenne de sécurité et de défense, Juin 2005, document A/1899.

[30] M. Corson and J. Macker, "Mobile ad hoc networking MANET: Routing protocol performance issues and evaluation considerations," RFC 2501, January 1999.

[31] Darpa, "DARPA Home Page," http://www.darpa.mil/.

[32] S. K. Das, B. S. Manoj, and C. S. R. Murthy, "A Dynamic Core Based Multicast Routing Protocol for Ad hoc Wireless Networks," in *3rd ACM MobiHOC 2002*, Lausanne, Switzerland, June 2002, pp. 24 – 35.

[33] A. D. de Clauzade de Mazieux, M. Marot, and M. Becker, "An analysis of the generalised Max-Min d-cluster formation heuristic," in *The Sixth Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net 2007)*, Ionian University, Corfu, Greece, June 2007.

[34] S. Deering, "Host extension for IP multicasting," RFC 1112, August 1989.

[35] S. Deering and D. Cheriton, "Multicast routing in datagram internetworks and extended LANs," *ACM Transcations on Computer Systems*, vol. 8, no. 2, pp. 85–110, May 1990.

[36] S. Deering, "Multicast routing in a datagram network," Ph.D. dissertation, Standford University, 1991.

[37] H. Dhillon and H. Ngo, "CQMP: a Mesh-based Multicast Routing Protocol with Consolidated Query Packets," in *IEEE Wireless Communications and Networking Conference (WCNC'05)*, vol. 4, March 2005, pp. 2168–2174.

[38] A. Ephremides, J. Wieselthier, and D. Baker, "A design concept for reliable mobile radio networks with frequency hoping signaling," in *IEEE 75*, 1987, pp. 56–73.

[39] H. Eriksson, "MBone: the Multicast Backbone," *Communications of the ACM*, vol. 37, pp. 54–60, August 1994.

[40] B. Fenner, M. Handley, H. Holbrook, and I. Kouvelas, "Protocol Independent Multicast - Sparse Mode (PIM-SM) : Protocol Specification," RFC 4601, August 2006.

[41] B. Fenner and D. Meyer, "Multicast Source Discovery Protocol MSDP," RFC 3618, October 2003.

[42] W. Fenner, "Internet Group Management Protocol, version 2," RFC 2236, November 1997.

[43] ——, "Domain Wide Multicast Group Membership Reports," IETF Internet Draft, August 1999.

[44] Y. Fernandes and D. Malkhi, "K-clustering in wireless ad hoc networks," in *ACM international workshop on Principles of mobile computing*, Toulouse, France, 2002, pp. 31–37.

[45] A. L. G. Rodolakis, A. Meraihi Naimi, "Multicast Overlay Spanning Tree Protocol for Ad Hoc Networks," in *WWIC*, Coimbra, Portugal, 2007.

[46] M. Gandhi, "A Holistic Networking Perspective on Mobile Tactical Networks for the Departement of the Defense," in *Software Defined Radio Technical Conference*, November 2005.

[47] J. Garcia-Luna-Aceves and E. Madruga, "The Core-Assisted Mesh Protocol," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 8, pp. 1380–1394, August 1999.

[48] Y. Ge, L. Lamont, and L. Villasenor, "Hierarchical OLSR - A Scalable Proactive Routing Protocol for Heterogeneous Ad Hoc Networks," in *WiMob 2005 Wireless and Mobile Computing*, vol. 3, August, Montreal, Canada 2005, pp. 17–23.

[49] M. Gerla and J. T.-C. Tsai, "Multicluster, mobile multimedia radio network," *ACM/Baltzer Journal of Wireless Networks*, vol. 1, no. 3, pp. 255–265, July 1995.

[50] M. Ghassemian, P. Hofmann, C. Prehofer, V. Friderikos, and H. Aghvami, "Performance Analysis of Internet Gateway Discovery Protocols in Ad Hoc Networks," in *WCNC'04*, 2004, pp. 120–125.

[51] C. Gui and P. Mohapatra, "Efficient Overlay Multicast for Mobile Ad Hoc Networks," in *IEEE Wireless Communications and Networking Conference (WCNC'03)*, vol. 2, March 2003, pp. 1118–1123.

[52] ——, "Scalable Multicasting in Mobile Ad Hoc Networks," in *INFOCOM 2004*, vol. 3, March 2004, pp. 2119–2129.

[53] P. Gupta and P. Kumar, "The capacity of wireless network," *IEEE Trans. On Information Theory*, vol. 46, no. 2, pp. 388–404, March 2000.

[54] A. Hamidian, "A Study of Internet Connectivity for Mobile Ad Hoc Networks in NS 2," Departement of Communication Systems, Lund Institute of Technology, Lund University, Master's thesis, January 2003.

[55] H. J. Hang and M. J. Lee, "A Multi-Source Multicast Routing Protocol for Ad Hoc Networks," *Lecture Notes in Computer Science Journal (LNCS)*, pp. 309–320, January 2002.

[56] J. Hawkinson and T. Bates, "Guideline for creation, selection, and registration of an Autonomous System (AS)," RFC 1930, March 1996.

[57] H. Holbrook and C. Cheriton, "IP Multicast Channels: Express Support for Large-Scale Single-Source Applications," in *SIGCOMM'99*, vol. 29, no. 4, September 1999, pp. 65–78.

[58] C.-C. Huang, M.-H. Guo, and R.-S. Chang, "A Weight-based Clustering Multicast Routing Protocol for Mobile Ad Hoc Networks," *International Journal of Internet Protocol Technology*, vol. 1, no. 1, pp. 10–18, 2005.

[59] A. Iwata, C. Chiang, G. Pei, M. Gerla, and T. Chen, "Scalable routing strategies for Ad Hoc Wireless Networks," *IEEE Journal on Selected area in Communications*, vol. 17, no. 8, pp. 1369–1379, August 1999.

[60] P. Jacquet, A. Laouiti, P.Minet, and L. Viennot, "Performance evaluation of multipoint relaying in mobile ad hoc networks," in *Networking 2002*, Pisa, 2002.

[61] P. Jacquet, P. Minet, P. Muhlethaler, and N. Rivierre, "Increasing reliability in cable-free Radio LANs: Low level forwarding in HIPERLAN," *Wireless Personal Communications*, pp. 51–63, January 1997.

[62] C. Jelger, T. Noel, and A. Frey, "Gateway and Address Autoconfiguration for IPv6 Ad Hoc Networks," IETF Internet Draft, draft-jelger-manet-gateway-autoconf-v6-02.txt, April 2004.

[63] J. Jetcheva and D. Johnson, "Adaptive Demand-Driven Multicast Routing in Multi-Hop Wireless Ad Hoc Networks," in *Second ACM Symposium on Mobile Ad Hoc Networking and Computing*, Long Beach, CA, USA, October 2001, pp. 33–44.

[64] L. Ji and M. Corson, "A lightweight adaptive multicast algorithm," in *IEEE GLOBECOM'98*, vol. 2, Sydney, Australia, November 1998, pp. 1036–1042.

[65] ——, "Differential Destination Multicast - A MANET multicast routing protocol for small groups," in *IEEE Infocom*, 2001, pp. 1192–1202.

[66] D. Johnson, Y. Hu, and D. Maltz, "The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4," RFC 4728, Fevrier 2007.

[67] J. Jubin and D. Tornow, "the DARPA Packet Radio Network protocol," in *IEEE*, vol. 75, no. 1, January 1987, pp. 21–32.

[68] L. Kleinrock and K. Stevens, "Fisheye: A Lenslike Computer Display Transformation," UCLA Computer Science Department, Tech. Rep., 1971.

[69] P. Krishna, N. H. Vaidya, m. Chatterjee, and D. K. Pradhan, "A cluster based approach for routing in dynamic networks," in *ACM SIGCOMM*, April 1997, pp. 49–65.

[70] S. Kumar, P. Radoslavov, D. Thaler, C. Aleattinoglu, D. Estrin, and M. Handley, "The MASC/BGMP Architecture for Inter-domain Multicast Routing," in *ACM SIGCOMM'98*, vol. 28, no. 4, Vancouver, September 1998, pp. 93–104.

[71] P. F. L. Barriere and L. Narayanan, "Robust Position-Based Routing in Wireless Ad Hoc Networks with Unstable Transmission Ranges," in *5 th International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications (DIAL-M)*, 2001, pp. 19–27.

[72] A. Laouiti, P. Jacquet, P. Minet, L. Viennot, T. Clausen, and C. Adjih, "Multicast Optimized Link State Routing," INRIA, HYPERCOM Team, Rocquencourt, Tech. Rep., February 2003.

[73] A. Laouti, P. Muhlethaler, A. Najid, and E. Plakoo, "Simulation Results of the OLSR Routing Protocol for Wireless Network," in *1st Mediterranean Ad-Hoc Networks workshop (Med-Hoc-Net)*, Sardegna, Italy, 2002.

[74] L. K. Law, S. Krishnamurthy, and M. Faloutsos, "On evaluating the trade-offs between broadcasting and multicasting in ad hoc networks," in *IEEE Military Communications Conference*, vol. 2, November 2004, pp. 799–804.

[75] J. Lee, D. Kim, J. J. Garcia-Luna-Aceves, Y. Choi, J. Choi, and S. Nam, "Hybrid Gateway Advertisement Scheme for Connecting Mobile Ad Hoc Networks to the Internet," in *IEEE Vehicular Technology Conference, VTC 2003-Spring*, vol. 1, April 2003, pp. 191–195.

[76] S. Lee and C. Kim, "NSMP: Neighbor Supporting Ad Hoc Multicast Routing Protocol," in *ACM MobiHOC 2000*, Boston, august 2000, pp. 37–50.

[77] S.-J. Lee, M. Gerla, and C.-C. Chiang, "On-Demand Multicast Routing Protocol," in *IEEE WCNC'99*, New Orleans, LA, September 1999, pp. 1298–1302.

[78] S. Lee, W. Su, J. Hsu, M. Gerla, and R. Bagrodia, "A performance comparison study of ad hoc wireless multicast protocols," in *IEEE Infocom 2000*, vol. 2, March 2000, pp. 565–574.

[79] W. Liao and M. Jiang, "Family Ack Tree (FAT): Supporting Reliable Multicast in Mobile Ad Hoc Networks," *IEEE Transaction on Vehicular Technology*, vol. 52, pp. 1675–1685, November 2003.

[80] C. R. Lin and M. Gerla, "Adaptive clustering for mobile wireless networks," *IEEE Journal of Selected Areas in Communications*, vol. 15, no. 7, pp. 1265–1275, 1997.

[81] H.-C. Lin and Y.-H. Chu, "A clustering technique for large multihop mobile wireless networks," in *Vehicular Tachnology Conference, VTC'00*, vol. 2, Tokyo, Japan, May 2000, pp. 1545–1549.

[82] J. Luo, P. T. Eugster, and J. Hubaux, "Route Driven Gossip: Probabilistic Reliable Multicast in Ad Hoc Networks," in *INFOCOM'03*, San Francisco, CA, March 2003, pp. 2229–2239.

[83] J. Macker, "ZSimplified Multicast Forwarding for MANET," Internet-Draft, IETF MANET WG, draft-ietf-manet-smf-07.txt, February 2008.

[84] J. Maker and I. Chakeres, "IETF Mobile Ad-hoc Networks (Manet) Working Group," Charter, [on line] http://www.ietf.org/html.charters/manet-charter.html.

[85] M. Maleki and M. Pedram, "Lifetime-aware Multicast Routing in Wireless Ad Hoc Networks," in *IEEE Wireless Communications and Networking Conference (WCNC'04)*, vol. 3, March 2004, pp. 1317–1323.

[86] D. Maltz, D. Johnson, J. Jetcheva, and Y. Hu, "A performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols," in *4-th ACM Annual International Conference on Mobile Computing and Networking, MOBICOM*, 1998, pp. 85–97.

[87] J.-P. Maulny, *La guerre en réseau au XXIieme siecle : Internet sur les champs de bataille*. Edition du Felin, 2006.

[88] J. McGee, M. Karir, and J. S. Baras, "Implementing Ad Hoc to Terrestrial Network Gateways," *Lecture Notes in Computer Science Journal (LNCS)*, vol. 2957, pp. 132–142, February 2004.

[89] P. Mohapatra and S. Krishnamurthy, *Ad Hoc Networks Technologies and Protocols*, P. Mohapatra and S. Krishnamurthy, Eds. Springer, 2005.

[90] H. Moustafa and H. Labiod, "A Performance Analysis of Source Routing-based Multicast Protocol (SRMP) Using Different Mobility Models," in *IEEE International Conference on Communications*, vol. 7, June 2004, pp. 4192–4196.

[91] H. Moustafa, "Routage unicast et multicast dans les reseaux mobiles ad hoc," Ph.D. dissertation, Ecole Nationale Superieure des Telecommunications, December 2004.

[92] J. Moy, "Multicast Extension to OSPF," RFC 1584, March 1994, proteon Inc.

[93] C. Murthy and B. Manoj, *Ad hoc Wireless Networks Architectures and Protocols*, P. H. C. Engineering and E. T. Series, Eds. Hardcover, May 2004, iSBN 013147023X.

[94] T. Ohta, T. Kawaguchi, and Y. Kaduda, "An Autonomous Clustering-based Hierarchical Multicast Routing for Mobile Ad Hoc Networks," *IEICE Transaction on Communications*, vol. E88-B, no. 12, pp. 4451–4461, December 2005.

[95] OPNET, "www.opnet.com."

[96] T. Ozaki, J. B. Kim, and T. Suda, "Bandwidth Efficient Multicast Routing Protocol for Ad Hoc Networks," in *International Conference on Computer Communications and Networks (ICCCN'99)*, October 1999, pp. 10–17.

[97] E. Pagani and G. P. Rossi, "Reliable Broadcast in Mobile Multihop Packet Networks," in *MOBICOM'97*, Budapest, Hungary, September 1997, pp. 34–42.

[98] ——, "An On-Demand Shared Tree with Hybrid State for Multicast Routing in Ad Hoc Mobile Wireless Networks," in *International Workshop on Parallel Processing*, Japan, September 1999, pp. 4–9.

[99] G. Pei, M. Gerla, and T.-W. Chen, "Fisheye State Routing in Mobile Ad Hoc Networks," in *2000 ICDCS Workshops*, Taipei Taiwan, April 2000, pp. D71–D78.

[100] C. Perkins, E. Belding-Royer, and S. Das, "Ad-hoc On-Demand Distance Vector (AODV) Routing," RFC 3561, July 2003.

[101] C. E. Perkins, *Ad Hoc Networking*, A.-W. Professional, Ed. , 2001.

[102] R. Ramanathan and J. Redi, "A Brief Overview of Ad Hoc Networks: Challenges and Directions," in *IEEE Communications Magazine*, vol. 40, no. 5, May 2002, pp. 20–22.

[103] T. M. Rasheed, U. Javaid, L. Reynaud, and K. Al Agha, "Adaptive Weighted Clustering for Large Scale Mobile Ad Hoc Networking Systems," in *Wireless Algorithms, Systems and Applications, WASA'06*, Xi'an, China, August 2006, pp. 206–216.

[104] P. Ratanchandani and R. Kravets, "A hybrid approach to internet connectivity for mobile ad hoc networks," in *WCNC'03*, vol. 3, March 2003, pp. 1522–1527.

[105] M. Richard and D. Roth, "How the Global Information Gird is transforming Communications for the Warfighter," *Transforming Communication MITRE's Advanced Technology Newsletter*, vol. 9, no. 2, pp. 8–10, Fall 2005.

[106] F. J. Ros and P. M. Ruiz, "Cluster-based OLSR Extension to Reduce Control Overhead in Mobile Ad Hoc Networks," in *IWCMC'07*, Honolulu, Hawaii, August 2007.

[107] M. Rosenschon, T.Manz, J. Habermann, and V. Rakocevic, "Gateway Discovery Algorithm for Ad Hoc Networks Using HELLO Messages," in *International Workshop on Wireless Ad Hoc Networks, IWWAN'05*, London, May 2005.

[108] E. M. Royer and C. Perkins, "Multicast operation of the on-demand distance vector routing protocol," in *ACM/IEEE International Conference on Mobile computing and networking.* Seattle, August 1999, pp. 207–218.

[109] P. M. Ruiz, F. J. Ros, and A. Gomez-Skarmeta, "Internet Connectivity for Mobile Ad Hoc Networks: Solutions and Challenges," *IEEE Communications Magazine*, pp. 118–125, 2005.

[110] P. Ruiz, G. Brown, and I. Groves, "Scalable Multicast Communications for Ad Hoc Extensions Attached to IP Mobile Networks," in *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC'02*, vol. 3, September 2002, pp. 1053–1057.

[111] M. Ryan and M. Frater, *Tactical Communications for the Digitized Battlefield*, Norwood, Ed. Artech House Publishers, June 2002.

[112] M. Saghir, T. C. Wan, and R. Budiarto, *Lecture Notes in Computer Science*. Bangkok, Thailand: Springer-Verlag, 2005, vol. 3837, ch. Load Balancing QoS Multicast Routing Protocol in Mobile Ad Hoc Networks, pp. 83–97.

[113] ——, "QoS Multicast Routing Based on Bandwidth Estimation in Mobile Ad Hoc Networks," in *International Conference on Computer and Communication Engineering, ICCCE'06*, vol. 1, Kuala Lumpur, Malaysia, May 2006, pp. 384–389.

[114] C. A. Santivanez, B. McDonald, I. Stavrakakis, and R. R. 2002., "On the scalability of ad hoc routing protocols," in *INFOCOM 2002*, vol. 3, 2002, pp. 1688–1697.

[115] SCILAB, Scilab, plateforme open source de calcul scientifique, http://www.scilab.org/.

[116] Séssion Nationale du Centre des Hautes Etudes de l'Armement, "L'impact du concept d'opérations réseaux centrées sur les capacités de notre futur appareil de défense," Rapport de groupe, 41ieme Session Juin 2005.

[117] C. Shen and C. Jaikaeo, "Ad Hoc Multicast Routing Algorithm with Swarm Intelligence," *Mobile Networks and Applications*, vol. 10, no. 1-2, pp. 47–59, February 2005.

[118] Y.-Y. Su, S.-F. Hwang, and C.-R. Dow, "An Efficient Multi-Source Multicast Routing Protocol in Mobile Ad Hoc Networks," in *11th IEEE International Conference on Parallel and Distributed Systems (ICPADS'05)*, vol. 1, 2005, pp. 8–14.

[119] K. Tang, K. Obraczka, S. J. Lee, and M. Gerla, "Reliable Adaptive Lighweight Multicast Protocol," in *IEEE International Conference on Communications (ICC'03)*, vol. 2, Los Angeles, CA, May 2003, pp. 1054–1058.

[120] H. Tebbe and A. Kassler, "QAMNet: Providing Quality of Service to Ad-hoc Multicast Enabled Networks," in *1st International Symposium on Wireless Pervasive Computing (ISWPC)*, no. 1-5, Thailand, January 2006.

[121] D. Thaler, "Interoperability Rules for Multicast Routing Protocols," RFC 2715, October 1999.

[122] C.-K. Toh, G. Guichal, and S. Bunchua, "ABAM : On-Demand Associativity-Based Multicast Routing for Ad Hoc Mobile Networks," in *VTC'00*, vol. 3, Boston, MA, USA, September 2000, pp. 987–993.

[123] R. Vaishampayan and J. Garcia-Luna-Aceves, "Efficient and Robust Multicast Routing in Mobile Ad Hoc Networks," in *IEEE International Conference on Mobile Ad hoc and Sensor Systems, MASS'04*, Fort Landerdale (FL), USA, October 2004, pp. 304–313.

[124] ——, "Robust Tree-based Multicasting in Ad Hoc Networks," in *IEEE International Conference on Performance, Computing and Communications*, 2004, pp. 647–652.

[125] K. Viswanath, K. Obraczka, and G. Tsudik, "Exploring Mesh and Tree-Based Multicast Routing Protocols for MANETs," *IEEE Transaction on Mobile Computing*, vol. 5, no. 1, pp. 28–42, January 2006.

[126] E. Vollset and P. Azhilchelvan, "A survey of Reliable Broadcast Protocols for Mobile Ad-hoc Networks," University of Newcastle upon Tyne, Tech. Rep. CS-TR-792, 2003.

[127] D. Waitzman, C. Partridge, and S. Deering, "Distance Vector Multicast Routing Protocol," RFC 1075, November 1998.

[128] R. Wakikawa, J. Malinen, C. E. Perkins, A. Nilsson, and A. Tuominen, "Global Connectivity for IPv6 Mobile Ad Hoc Networks," IETF Internet Draft, draft-wakikawa-manet-globalv6-03.txt, October 2003.

[129] B. Wang and S. K. S. Gupta, "On Maximizing Lifetime of Multicast Trees in Wireless Ad Hoc Networks," in *International Conference on Parallel Processing (ICPP'03)*, October 2003, pp. 333–340.

[130] C. Wu and Y. Tay, "AMRIS: A multicast protocol for ad hoc wireless networks," in *IEEE MILCOM'99*, no. 1, Atlantic City,NJ, October 1999, pp. 25–29.

[131] S. Wu and C. Bonnet, "Multicast Routing Protocol with Dynamic Core," in *International Symposium on Telecommunications, IST'01*, September 2001, pp. 274–280.

[132] L. Xiao, A. Patil, Y. Liu, L. M. Ni, and A. Esfahanian, "Prioritized Overlay Multicast in Mobile Ad Hoc Environments," *IEEE Computer Magazine*, pp. 67–74, February 2004.

[133] Z. Xiaofeng and L. Jacob, "Multicast Zone Routing Protocol in Mobile Ad Hoc Wireless Networks," in *IEEE International Conference on Local Computer Networks LCN'03*, October 2003, pp. 150–159.

[134] J. Xie, S. Nandi, A. Gupta, and A. Das, "Gateway-based Multicast Protocol-A Novel Multicast Protocol for Mobile Ad Hoc Networks," *IEEE Proceedings Communications*, vol. 152, no. 6, pp. 811–820, December 2005.

[135] Y. Yi, M. Gerla, and K. Obraczka, "Scalable Team Multicast in Wireless Ad Hoc Netkworks Exploiting Coordinated Motion," *Ad Hoc Networks*, vol. 2, no. 2, pp. 171–184, April 2004.

[136] Y. Yi, S.-J. Lee, W. Su, and M. Gerla, "On-Demand Multicast Routing Protocol (ODMRP) for ad hoc networks," Internet Draft, expired, draft-yi-manet-odmrp-00, March 2003.

[137] J. Yu and P. Chong, "A survey of clustering schemes for mobile ad hoc networks," *IEEE Communications Surveys & Tutorials*, vol. 7, no. 1, pp. 32–48, First Qtr. 2005.

[138] Y. Zhao, L. Xu, and M. Shi, "On-Demand Multicast Routing Protocol With Multipoint Relay (ODMRP-MPR) in Mobile Ad Hoc Network," in *International Conference on Communication Technology (ICCT'03)*, vol. 2, 2003, pp. 1295–1300.

Résumé Long En Français

A.1 Introduction

Depuis les 20 dernières années, l'émergence de nouvelles technologies d'information et de communications a radicalement transformé la société. Inspiré par cette révolution qui nous a amené aujourd'hui à ce qu'on appelle "l'Age de l'information", les concepts militaires opèrent une Transformation, passant d'une force orientée plateforme à une force orientée réseau. Ce concept d'opérations orientées réseau place l'information, dans le sens d'intelligence stratégique, opérationnelle et tactique, au premier rang des problématiques de tous les DoD.

Dans cette partie introductive, nous présentons le contexte global dans lequel s'inscrit ce travail de thèse. Pour cela nous nous proposons de décrire la nouvelle architecture des communications dans le domaine militaire afin de faire apparaitre les nouveaux besoins en termes de réseaux sans fil. Ainsi, dans une première partie, nous présentons le concept de "Network Centric Warfare" (NCW) et son implémentation via le "Global Information Grid" (GIG). Ensuite, étant donné que le GIG nécessite des modifications de l'architecture traditionnelle des communications militaires, nous présentons cette nouvelle architecture en nous concentrant plus particulièrement sur les communications tactiques. Le concept de réseau mobile Ad Hoc (Mobile Ad Hoc Network - MANET-) occupe une place centrale dans la future architecture des communications tactiques. Ainsi, nous décrivons dans une troisième partie les spécificités d'un réseau tactique MANET. A l'issue de ces trois parties, le contexte de notre travail étant clairement défini, nous présentons les objectifs et l'organisation de cette thèse.

Le concept de "Network Centric Warfare" est apparu en 1998 aux Etats Unis comme une nouvelle façon de penser les opérations militaires à l'Age de l'information. La volonté était d'utiliser les nouvelles technologies du domaine des réseaux de communications émergeantes dans le monde commercial. Le concept de NCW peut donc être défini comme la mise en réseau des systèmes de commandes, de contrôles et d'armes, i.e. tous les acteurs du monde militaire grâce aux Nouvelles Technologies de l'Information et de la Communication (NTIC). Ainsi cette nouvelle vision amène à abandonner la structure hiérarchique des communications militaires au profit d'une architecture de communication plus réactive et plus efficace où tous les acteurs (du commandement au soldat sur le terrain) sont connectés de façon permanente, en temps réel et où tous les théâtres opérationnels sont mis en réseau.

Dans cette thèse nous nous intéressons plus particulièrement aux communications dites tactiques, c'est-à-dire les communications entre acteurs de la région opérationnelle tactique, du champ de bataille opérationnel. Traditionnellement, les communications dans cet environnement sont supportées par une architecture hiérarchique où un premier sous-système (Trunk communication subsystem) permet d'interconnecter les quartiers généraux avec les niveaux brigade et supérieurs, un deuxième sous-système (Combat Net Radio) fourni le support pour les communications entre les troupes de combat aux niveaux brigade et inférieurs et un troisième sous-système (Local Area Network) est utilisé pour les communications dans les quartiers généraux et entre les différents véhicules à l'arrêt. Avec cette architecture, le réseau est statique, peu évolutif car chacun ne peut que garder la place qu'on lui a prédéfini par avance, les communications sont point-à-point et finalement, les noeuds de communications critiques sont définis à l'avance et doivent fonctionner à l'arrêt. L'introduction du concept de Network Centric Warfare permet de "casser" cette architecture hiérarchique puisqu'il nécessite de fournir des capacités de communication transparentes, par dessus les frontières traditionnelles, dans des délais déterministes, du déploiement jusque sur le champ de bataille et ce, avec des acteurs en mouvement constant.

Ainsi, le concept de Réseau Tactique, aussi appelé Internet Tactique, est apparu pour remplacer les systèmes de communications tactiques traditionnels. Au sein de ce réseau, on trouve l'Internet Tactique Mobile composé de Noeuds de Communications Tactiques (TCN) qui fournissent des nouvelles capacités radio pour interconnecter les troupes de combat qui ont une forte dynamique. Un TCN est une plateforme véhiculaire qui intègre des équipements radio et des fonctions réseaux. Un ensemble de TCNs forment un réseau radio très dynamique (le nombre et le type de noeuds varient de façon imprédictible), autoconfigurable, mobile, qui doit pouvoir être déployé n'importe où et n'importe quand en fournissant des capacités de communications même quand les noeuds sont mobiles.

Ainsi, à la vue de ces caractéristiques, le concept de réseau Mobile Ad Hoc (MANET) apparait comme

totalement adapté pour satisfaire les contraintes demandées. Ainsi, par la suite, l'équipement radio du TCN sera appelé noeud MANET et l'ensemble de ces équipements radio formeront le réseau MANET tactique. Nous ne rappelons pas dans ce résumé l'historique et les caractéristiques d'un réseau MANET. Nous donnons par contre, les spécificités d'un réseau MANET dans son emploi Tactique :

- **Intégration et Interopérabilité**: l'Internet Tactique est composé de plusieurs sous systèmes, parmi lesquels le MANET tactique, qui doivent tous être intégrés ensemble. Ainsi, le MANET tactique doit être interopérable avec des réseaux sans fil ou filaire commerciaux, avec des réseaux militaires de pays différents, avec des réseaux militaires différents, etc ... De plus, le MANET tactique est utilisé en tant que réseau de transit contrairement à l'utilisation traditionnelle où le MANET est un réseau de bout. Ainsi, un noeud du MANET tactique a principalement le rôle de routeur et non d'hôte et routeur, et permet donc de faire transiter le trafic généré par des hôtes présents sur d'autres réseaux connectés au noeud MANET. En tant que réseau de transit, le MANET tactique doit donc s'interconnecter avec d'autres réseaux utilisant potentiellement d'autres protocoles pour rendre les services de routage, de QoS, de sécurité...

- **Passage à l'échelle**: l'utilisation du MANET dans le domaine tactique a un impact sur la taille des réseaux à considérer. Ainsi, le MANET tactique peut atteindre une taille de plusieurs milliers de routeurs. Ce type de déploiement est très différent de ceux rencontrés dans le domaine commercial où la taille des réseaux MANET n'est pas supposée dépasser quelques centaines de noeuds. De plus, un MANET tactique doit pouvoir être utilisé dans une vaste gamme de déploiements opérationnels où la densité peut varier de faible à très importante, où la mobilité peut aller de la vitesse un soldat à pied à celle d'un véhicule héliporté.

- **Unicast**: la majorité des communications dans l'environnement tactique sont de profil point-à-point. Ainsi un protocole de routage unicast adapté au MANET doit être implémenté. Ce domaine de recherche est très abondant, si bien qu'un groupe de travail IETF a été créé pour adresser ce problème. Cependant, le problème de la scalabilité des protocoles de routage est toujours un problème ouvert.

- **Multicast**: beaucoup d'applications militaires comme le "situational awarness", la collaboration, la vidéoconférence, la voie groupée, la gestion du réseau sont des applications de groupes nécessitant la présence d'un service multicast. Ce type de service doit être étudié avec une grande attention dans le contexte des MANET tactiques car une part importante des données militaires sera de nature multicast.

L'objectif de cette thèse est donc de prendre en compte les spécificités des réseaux MANET tactiques de façon à définir l'architecture de communications et les protocoles nécessaires pour fournir un service multicast à l'environnement tactique via le MANET tactique. L'architecture réseau et notamment le fait que le MANET tactique est un réseau de transit devra être prise en compte dans cette étude. De plus, même si l'objectif principal est d'étudier le service multicast, il ne faut pas oublier que celui-ci est intégré dans un environnement où d'autres services doivent être rendu et ce de façon optimale.

L'organisation de la thèse est la suivante. Dans une deuxième partie, l'architecture protocolaire du service multicast dans le réseau tactique est étudiée. Cette étude permet de mettre en évidence la nécessité d'un protocole de routage multicast spécifique dans le MANET tactique. De plus, la contrainte de passage à l'échelle vis-à-vis du nombre de noeuds souligne la nécessité de définir une solution basée sur le clustering qui est la solution communément retenue pour supporter le passage à l'échelle dans les MANET. Utiliser un protocole de clustering qui divise le réseau en groupes de noeuds amène à considérer deux niveaux de communications multicast en fonction de la répartition des acteurs des communications de groupes sur les clusters. Le premier niveau de communication se réfère aux communications multicast internes à un cluster. Ce problème est traité dans la troisième partie. Après un état de l'art de l'existant, le protocole STAMP (Shared Tree Ad hoc Multicast Protocol) ainsi que les résultats de l'étude de performance réalisée sur ce protocole sont proposés. Le deuxième niveau de communications multicast se réfère aux communications multicast entre les clusters. Ce problème est traité dans la quatrième partie. Apres un état de l'art de l'existant dans le domaine des protocoles de routage multicast dans les réseaux MANET clustérisés, le protocole SAFIR (ScAlable structure-Free Inter-cluster multicast Routing) ainsi que l'évaluation de performance réalisée sur celui-ci sont présentés. La cinquième partie traite des problèmes d'interopérabilité des protocoles multicast définis pour l'environnement du MANET tactique avec les protocoles de routage multicast potentiellement différents déployés sur les réseaux interconnecté via le MANET. Les protocoles multicast proposés au cours de cette thèse étant directement liés aux protocoles de routage unicast sous-jacents, nous proposons, dans une sixième partie, une solution de routage unicast reposant sur OLSR et supportant le passage à l'échelle. Enfin, la dernière partie conclue la thèse et propose des perspectives de travail à suivre pour continuer le travail entrepris au cours de ces trois années.

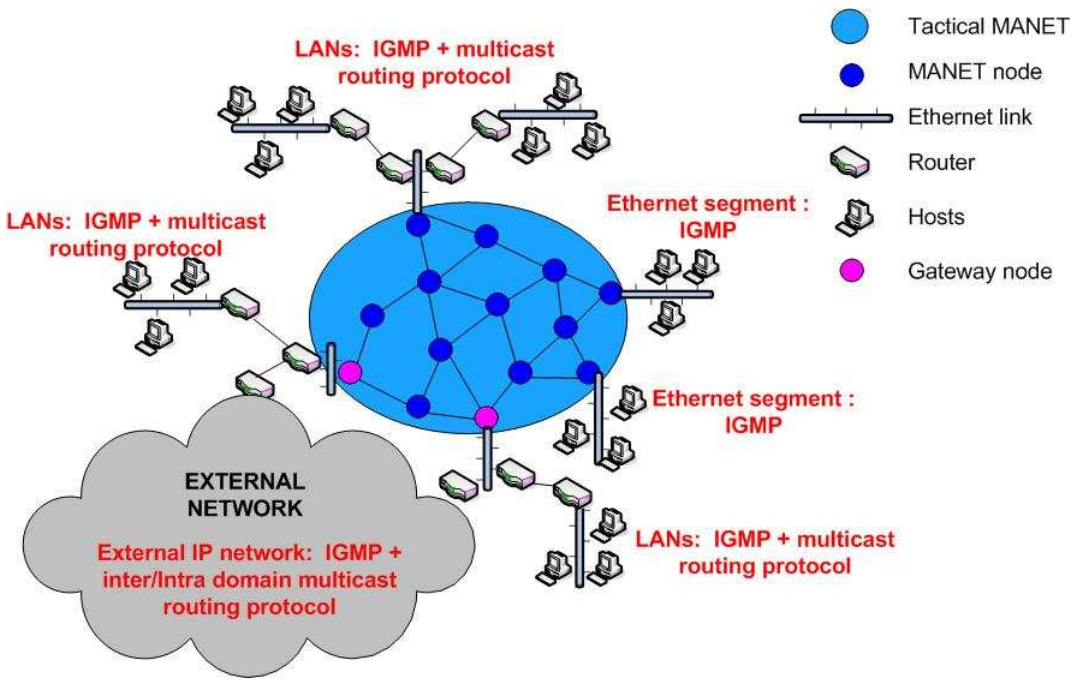

Figure A.2 Architecture du réseau d'un point de vue du service multicast

A.2 Architecture du Service de Communication Multicast

Cette partie étudie l'architecture du service de communication multicast qui doit être mise en place de façon à pouvoir gérer de façon transparente des communications multicast de bout-en-bout au travers du MANET tactique. Dans le manuscrit, le multicast IP pour les réseaux filaire est présenté (le modèle de communication multicast Internet, le protocole IGMP pour la gestion des appartenances aux groupes et les protocoles de routage multicast). La figure A.2 présente la structure du réseau Internet Tactique d'un point de vue du service multicast. Ainsi, le MANET tactique permet d'interconnecter différents réseaux de l'Internet Tactique que l'on peut classer en trois catégories en fonction du type de structure protocolaire mise en place pour le service multicast. Tout d'abord, on trouve des segments Ethernet qui sont composés exclusivement d'hôtes directement connectés au noeud MANET et qui implémentent le protocole IGMP pour reporter les appartenances aux groupes multicast. Ensuite, on trouve des LAN locaux qui sont composés d'hôtes employant le protocole IGMP et de routeurs employant un protocole de routage multicast. Le noeud MANET peut être relié à ces routeurs ainsi qu'aux hôtes. Enfin, on trouve ce que l'on va appeler des Réseaux IP Externes qui peuvent être n'importe quel type de réseau (l'Internet global, un réseau militaire IP, ...). Pour ce type de réseau le noeud MANET est relié à des routeurs qui déploient des protocoles de routage multicast intra et/ou inter domaine. Bien que les deux derniers types de réseaux puissent sembler redondants, une distinction doit être faite car contrairement aux LAN locaux qui peuvent être considérés comme des réseaux "propriétaires", les Réseaux IP Externes sont des réseaux totalement ouverts sur lesquels aucun contrôle, aucune hypothèse ou aucune recommandation ne peut être fait. Etant donné que les acteurs des sessions multicast (sources ou membres) peuvent se trouver sur n'importe lequel de ces trois types de réseau, le MANET peut être utilisé pour interconnecter différents types de protocole multicast. Nous rappelons que concernant le service multicast, un noeud MANET n'est en aucun cas responsable de décider d'appartenir à un groupe multicast ou d'être source de trafic pour un groupe multicast, ce sont les hôtes présents sur un des trois types de réseaux identifiés précédemment qui sont les réels acteurs des groupes multicast.

Trois solutions peuvent être envisagées pour fournir un service multicast de bout en bout :

- La première solution consiste à considérer que les noeuds MANET sont des routeurs n'implémentant aucun protocole multicast. On va alors créer des tunnels sur le MANET tactique à la manière de ce qui peut être fait sur le MBone. Cette solution présente des problèmes de performances notamment vis-à-vis de l'overhead de contrôle.

- La deuxième solution consiste à déployer sur les noeuds MANET les protocoles de routage multicast

définis pour les réseaux IP filaires. Malheureusement, ces protocoles sont peu adaptés aux contraintes du monde sans fil et à la mobilité.

- La troisième solution, qui est celle retenue, consiste à déployer sur les noeuds MANET un protocole de routage multicast spécialement conçu pour l'environnement MANET tactique, et ensuite à mettre en place des solutions d'interopérabilité avec les protocoles déployés sur les réseaux IP reliés au MANET tactique.

Ce protocole de routage multicast spécifiquement conçu pour le domaine du MANET tactique doit répondre aux contraintes imposées par cet environnement, à savoir :

- Robustesse: le protocole doit être capable de réagir aux changements de topologie du réseau causés par la mobilité des noeuds, et ce, en minimisant les pertes d'information.

- Efficacité: la bande passante est une ressource très rare dans les réseaux MANET. Ainsi, l'efficacité qui peut se mesurer comme le ratio du nombre total de paquets de contrôle et de données transmis sur le réseau sur le nombre total de paquets de données reçus doit être optimisée.

- Overhead de contrôle réduit: ce point rejoint le point précédent. Ainsi pour construire et maintenir la structure nécessaire à son fonctionnement, le protocole de routage doit réduire au maximum les messages de contrôle nécessaires de façon à ce qu'ils n'occupent pas toute la bande passante disponible, empêchant ainsi toute transmission de trafic de données.

- Qualité de service.

- Gestion des ressources: le protocole doit réduire au maximum l'utilisation des ressources mémoire, batterie, ... des noeuds. Les opérations nécessitant le plus de ressources doivent être réduite.

- Sécurité

- Passage à l'échelle: comme expliqué précédemment, le nombre de noeuds dans un réseau MANET tactique peut atteindre plusieurs milliers.

Il est bien évidemment difficile de concevoir un protocole répondant parfaitement à toutes ces exigences. La plus restrictive et contraignante des exigences présentées précédemment est celle de passage à l'échelle. En effet, la façon dont le passage à l'échelle est pris en compte dans un MANET peut influencer la structure du réseau et donc la conception d'une solution multicast.

Nous avons choisi le clustering comme solution pour gérer le passage à l'échelle car elle nous apparaît comme la plus prometteuse et la plus capitalisable possible. En effet, le clustering est une technique qui permet de regrouper les noeuds en groupes appelés cluster et d'assigner aux noeuds des fonctions et des responsabilités différentes selon qu'ils sont dans ou en dehors d'un groupe. Un noeud particulier dans chaque cluster, le clusterhead, est chargé de "représenter" les noeuds du cluster pour les opérations dépassant les frontières du cluster. Cela permet de réduire la sphère de responsabilité de chaque noeud ainsi que sa vision du réseau. Les avantages du clustering s'appliquent non seulement aux protocoles de routage multicast mais aussi aux protocoles de routage unicast, aux protocoles de gestion du réseau, à la réutilisation des fréquences, ... Par la suite, nous considèrerons donc le MANET tactique comme un réseau clustérisé comme illustré par la figure A.3. Cette structure clustérisée a une influence sur le service multicast dans le MANET. En effet, la répartition des acteurs parmi les différents clusters amène à considérer deux niveaux de communications multicast.

- Si les acteurs d'un groupe multicast appartiennent au même cluster, on parlera de communications multicast intra-cluster.

- Si les acteurs d'un groupe multicast sont répartis sur différents clusters, on parlera de communications multicast inter-cluster.

Pour tirer parti de la structure en cluster de façon optimale, on devra donc définir deux types de protocole de routage multicast, un protocole multicast intra-cluster qui se chargera des communications multicast intra-cluster et un protocole de routage multicast inter-cluster qui se chargera des communications multicast inter-cluster. Ces deux protocoles coopèreront de façon à fournir un service multicast de bout-en-bout.

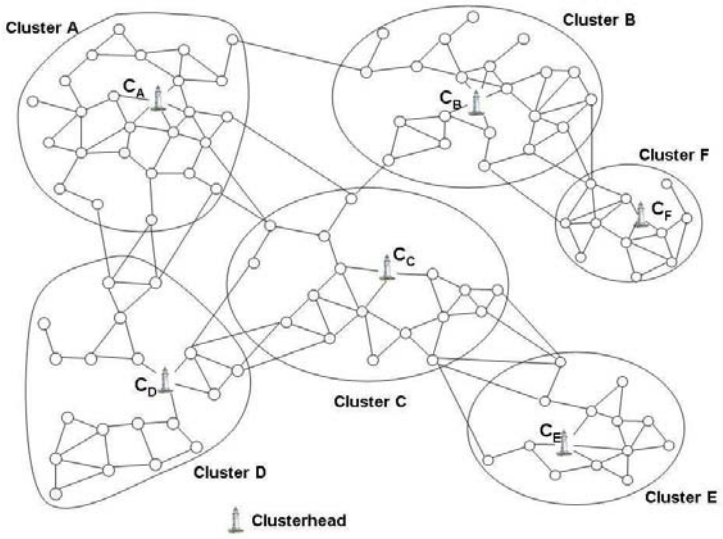

Figure A.3 Illustration d'un MANET tactique clustérisé

A.3 Protocole de routage multicast Intra-cluster

Cette partie adresse la problématique des communications multicast internes à un cluster. On suppose que la taille des clusters résultants du fonctionnement du protocole de clustering permet de considérer que les protocoles "non hiérarchique" défini pour les réseaux MANET commerciaux peuvent répondre à la problématique de l'intra-cluster. Ainsi, un état de l'art des solutions de multicast proposées au cours des dix dernières années doit être effectué afin de déterminer si une des solutions existantes permet de répondre aux exigences énoncées précédemment.

Afin de classer la multitude de solutions proposées dans la littérature, plusieurs taxonomies se fondant sur une des caractéristiques du protocole ont été envisagées, dont la topologie de la structure, le schéma d'acquisition des routes, l'initialisation de la session multicast, la dépendance vis-à-vis du protocole de routage unicast, le mécanisme de maintenance de la topologie ou la connectivité de la structure.

Le tableau A.1 donne une classification de quelques protocoles en fonction de ces différentes taxonomies. Malheureusement, ces taxonomies qui se concentrent sur une caractéristique précise du protocole ne permettent pas de conclure quant à l'existence d'un protocole pouvant répondre à nos exigences de robustesse, efficacité et économie d'énergie. Seule la taxonomie basée sur la structure présente un plus grand intérêt car c'est celle qui permet d'avoir la vue la plus globale sur le protocole. Cependant, cette taxonomie ne permet pas de prendre en compte les dernières approches émergeantes dans le domaine du multicast pour les réseaux MANET. Ces approches sont différentes dans le sens où soit elles exploitent une des caractéristiques du réseau MANET, soit elles se concentrent sur un objectif de design comme l'économie d'énergie par exemple. Afin d'avoir une meilleure comparaison des solutions existantes en ayant comme point de mire notre objectif, nous proposons une nouvelle taxonomie qui sera plus globale et plus "pratique" dans le sens où les protocoles seront classés en fonction de l'objectif de design (ou de l'exigence) qu'ils privilégient. On distingue les protocoles qui privilégient la robustesse, ceux qui privilégient l'efficacité et ceux qui privilégient l'économie d'énergie. Il ressort de cette étude qu'il n'existe pas de protocole qui permet de satisfaire en même temps les exigences de robustesse, d'efficacité et d'économie d'énergie. Ainsi nous avons choisi de définir un nouveau protocole qui combine les avantages des protocoles robustes et ceux des protocoles efficace. Concernant l'économie d'énergie, nous considérons qu'un protocole efficace, i.e. un protocole qui réduira au maximum les overheads de contrôle et de donnée, répondra aussi à l'exigence d'économie d'énergie dans le sens où la transmission et la réception sont deux des opérations les plus consommatrice en énergie.

Le protocole "Shared Tree ad Hoc Multicast Protocol" (STAMP) a été défini avec l'objectif d'être à la fois robuste et efficace. Ainsi, il repose sur une structure d'arbre partagé qui permet de réduire au minimum la surcharge d'information de contrôle ainsi que la duplication des données. De plus, il utilise une approche "hard state" pour la maintenance, i.e. les états de la structure multicast ne sont remis à jour que sur détection d'une rupture de liens et non périodiquement, de façon, ici encore, à réduire la surcharge d'information de contrôle. d'autre part, il tire parti de la capacité de diffusion du médium sans fil pour distribuer les données sur l'arbre comme sur un mesh de façon à créer de la redondance et ainsi augmenter la robustesse du protocole. Pour cela, il suffit de modifier la règle d'acceptation d'un paquet de données. Ainsi, au lieu de n'accepter un paquet de donnée en réception que s'il provient d'un noeud appartenant à la même branche de l'arbre (noeud

Table A.1 Tableau comparatif des différents protocoles de routage multicast

	Topology	Route Acquisition	Initialization	Unicast Dependency	Topology Maintenance	Structure Connectivity
ABAM [122]	Source Tree	Reactive	Traffic	Independent	Hard State	Source
ADMR [63]	Source Tree	Reactive	Traffic	Independent	Soft State	Source
AMRIS [130]	Shared Tree	Reactive	Elected Source	Independent	Soft State	Group
AMRoute [12]	Shared Tree	Proactive	Both	Partially Dependent	Soft State	Group
ASTM [24]	Shared/Source Tree	Proactive	Both	Partially Dependent	Soft State	Source and Group
BEMR [96]	Mesh	Proactive	Both	Independent	Hard State	Group
CAMP [47]	Mesh	Proactive	Both	Partially Dependent	Soft State	Group
CQMP [37]	Mesh	Reactive	Source	Independent	Soft State	Group
DCMP [32]	Mesh	Reactive	Source	Independent	Soft State	Group
DDM [65]	Source Tree	Reactive	Source	Dependent	None	Source
DPUMA, PUMA [123]	Mesh	Proactive	Group	Independent	Soft State	Group
FGMP-SA [25]	Mesh	Proactive	Source	Partially Dependent	Soft State	Source
FGMP-RA [25]	Mesh	Proactive	Group	Partially Dependent	Soft State	Group
GBMP [134]	Shared Tree	Reactive	Source	Independent	Soft State	Group
LAM [64]	Shared Tree	Proactive	Both	Partially Dependent	Hard State	Group
MANSI [117]	Mesh	Reactive	Source	Independent	Soft State	Group
MAODV [108]	Shared Tree	Proactive	Both	Independent	Soft State	Group
MOLSR [72]	Source Tree	Reactive	Source	Independent	Soft State	Source
MRDC [131]	Shared Tree	Reactive	First Source	Independent	Soft/Hard State	Group
MSTP [98]	Shared Tree	Hybrid	Source	Independent	Hybrid	Group
MZRP [133]	Source Tree	Hybrid	Source	Independent	Soft State	Source
NSMP [76]	Mesh	Reactive	Source	Independent	Soft State	Group
ODMRP [77]	Mesh	Reactive	Source	Independent	Soft State	Group
ODMRP-MPR [138]	Mesh	Reactive	Source	Independent	Soft State	Group
ODMRP-PDA [14]	Mesh	Reactive	Source	Independent	Soft State	Group
ROMANT [124]	Shared Tree	Proactive	Group	Independent	Soft State	Group
SMMRP [55]	Mesh	Reactive	Source	Independent	Soft/Hard State	Group
SRMP [90]	Mesh	Reactive	Source	Independent	Soft State	Group

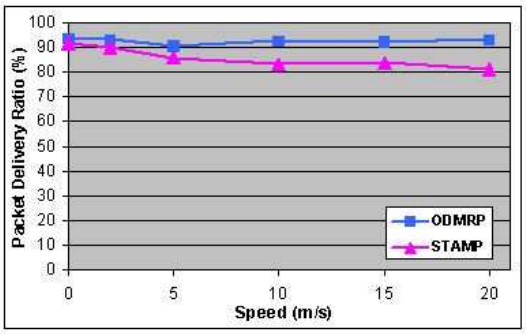

Figure A.4 Taux de délivrance des paquets en % en fonction de la mobilité.

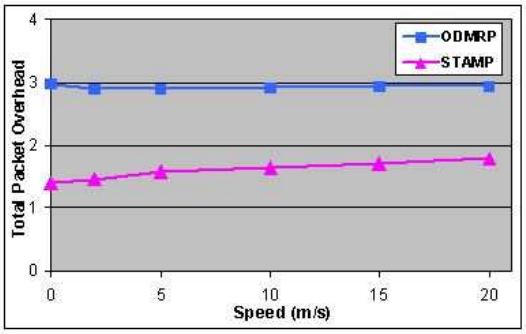

Figure A.5 Overhead de paquets total (données + contrôle) en fonction de la mobilité

identifié comme étant père ou fils sur la structure), un noeud de l'arbre acceptera en réception un paquet de donnée quelque soit le voisin qui l'a envoyé (à condition qu'il ne l'a pas déjà reçu bien évidemment). Le noeud qui sera le coeur de l'arbre sera le premier noeud à joindre le groupe multicast. Ce noeud envoie périodiquement un message d'annonce du coeur pour informer les autres noeuds du réseau. Dans l'utilisation du protocole en tant que protocole intra-cluster ces dernières règles ne s'appliquent pas, car le noeud coeur de l'arbre est déjà identifié et connu de tous les noeuds, c'est le clusterhead.

Nous avons effectué une étude d'évaluation de performance grâce au simulateur à évènement discret OPNET Modeler 11.5. L'objectif de cette étude était de vérifier que le protocole STAMP répondait bien aux exigences que nous nous étions fixées en début de conception. Ainsi, nous avons pris comme référence le protocole ODMRP, le plus représentatif des protocoles multicast basés sur un mesh, qui sont les protocoles reconnus pour être les plus robustes. Nous avons considéré différents types de scénarios pour évaluer le protocole en fonction de différentes configurations réseau. Ainsi, nous avons étudié l'influence de la mobilité des noeuds, du nombre de sources, du nombre de membres multicast par groupe, de la charge de trafic et de la densité du réseau. Nous avons pu montrer que STAMP atteint des taux de délivrance des données comparable à ceux d'ODMRP (cf. figure A.4), ce qui témoigne de sa robustesse. De plus, nous avons aussi montré que ces résultats étaient obtenus avec une efficacité beaucoup plus importante que celle d'ODMRP (cf. fig A.5). Les courbes ci-dessous donnent des exemples de résultats obtenus sur les scenarios de mobilité. La figure A.6 présente une comparaison différentielle en pourcentage des overhead de données (bleu), de contrôle (vert) et du taux de délivrance des paquets (rouge) en fonction de la mobilité. Elle illustre la faible perte en termes de robustesse et l'important gain en termes d'efficacité. Considérons par exemple les résultats obtenus pour la vitesse de 2m/s, on peut voir que STAMP produit 98% d'overhead de contrôle de moins qu'ODMRP, 36% d'overhead de données de moins et que son taux de délivrance des paquets est seulement de 3% inférieur.

A.4 Protocole de routage multicast Inter-cluster

Cette partie adresse la problématique des communications multicast entre les différents clusters. En effet, si l'on considère une répartition des membres de groupe multicast comme le groupe 2 sur la fig. A.7, le protocole STAMP permettra de gérer la distribution des données multicast entre les membres d'un même cluster comme

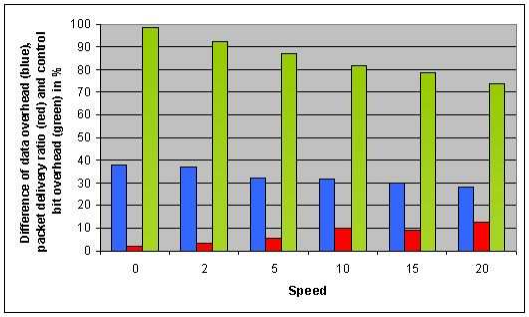

Figure A.6 Comparaison différentielle des overheads de données (bleu), de contrôle (vert) et du taux de délivrance des paquets (rouge) en fonction de la mobilité.

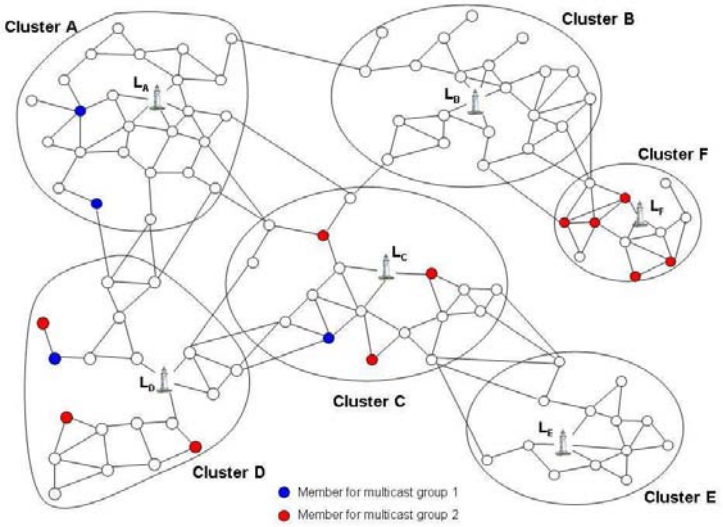

Figure A.7 Illustration de la répartition des membres de groupes multicast sur le réseau MANET clustérisé

dans les clusters F, C ou D et le protocole de routage multicast inter-cluster se chargera d'acheminer les données multicast entre ces différents clusters. Si la littérature est très abondante concernant les protocoles de routage multicast pour des environnements réseau non clustérisés avec un nombre de noeuds réduit, ce n'est pas du tout le cas concernant les protocoles de routage multicast pour les environnements réseau clustérisés. Une étude de l'existant nous permet cependant de distinguer deux catégories de protocoles. La première catégorie se compose de protocoles qui ne considèrent pas le réseau comme étant clustérisé mais qui crée une structure multicast sur le réseau global pour ensuite la diviser en sous structures gérées localement. Cette approche n'est pas compatible avec notre approche car elle ne tire pas partie de la structure clustérisée du réseau et elle ne fait pas de distinction entre les niveaux de routage inter et intra cluster. La seconde catégorie est en phase avec notre approche car elle consiste en des protocoles qui ont été conçu pour adresser le routage multicast inter-cluster seulement. Ces protocoles sont tous similaires dans le sens où ils proposent d'appliquer les solutions définies pour les réseaux "à plat" au niveau de la topologie des clusters. Cette solution est assez peu efficace du fait de l'effet "multiplicatif" du clustering sur les messages de contrôle et sur les erreurs. De plus, cette approche ne tire pas partie du fait qu'il existe dans un réseau clustérisé d'autres services (tels que le routage unicast, le clustering, ...) qui échangent eux aussi des messages de contrôle dont on peut tirer partie. En effet, le protocole de routage multicast inter-cluster pourrait faire du "piggybacking" de façon à intégrer ces informations de contrôle dans les messages de contrôle existants. En conclusion de cet état de l'art, nous concluons qu'il n'existe pas de solution suffisamment optimisée et intégrée pour le routage multicast inter-cluster dans un environnement MANET. Nous proposons donc le protocole SAFIR (ScAlable structure Free inter-cluster multicast Routing).

SAFIR défini une méthode pour router des datagrammes multicast entre clusters dans un réseau clustérisé fait de noeuds sans fil mobiles. SAFIR permet de gérer les communications inter-cluster et suppose

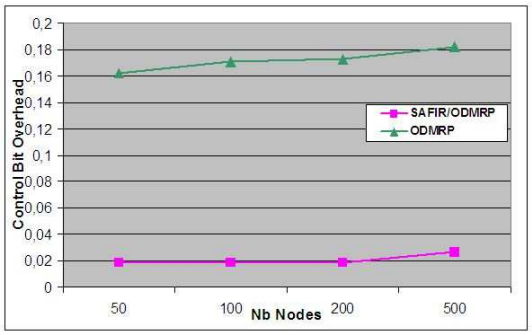

Figure A.8 Overhead de contrôle en bit en fonction du nombre de noeuds

qu'un protocole de routage intra-cluster tel que STAMP est implémenté pour gérer les communications multicast à l'intérieur de chaque cluster. Ainsi, l'objectif de SAFIR est de définir comment un datagramme pour un groupe multicast G peut être acheminé de cluster à cluster jusqu'à atteindre les clusters où se trouvent les membres du groupe multicast G. Cet objectif permet de mettre en avant différentes questions auxquelles SAFIR doit répondre :

1. Comment un noeud connaît-il la liste des clusters où se trouvent les membres pour un groupe multicast G?

2. Comment un cluster connaît-il vers quel cluster voisin il doit faire suivre les datagrammes multicast?

3. Quel noeud est responsable de prendre la décision de "forwarding"?

4. Sur quelle information se fonde la décision de forwarding?

5. Comment est faite l'interconnexion entre le protocole inter-cluster et le protocole intra-cluster? Quelles sont les informations qui doivent être échangées?

Dans SAFIR, comme dans la plupart des protocoles de routage multicast inter-cluster, c'est le clusterhead qui a la responsabilité de décider si un datagramme multicast doit être forwardé ou non. Contrairement aux autres protocoles de la littérature, aucun message Join / Leave / Ack n'est échangé entre les clusterheads ou les noeuds gateway pour construire ou maintenir une structure multicast qui permettrait aux clusterheads de prendre leur décision de forwarding. SAFIR défini une méthode dans laquelle chaque clusterhead prend sa décision de forwarding indépendamment des autres clusters, de façon autonome, de telle façon que chaque paquet multicast suit son propre chemin quand il va de cluster en cluster. Pour cela, chaque clusterhead doit connaître deux informations sur son cluster. d'une part, il doit savoir quels sont les groupes multicast pour lesquels il y a des membres dans son cluster et d'autre part, il doit connaître la liste des ses clusters voisins. Ces deux informations sont échangées entre les clusterheads en utilisant les messages de contrôle du protocole de clustering ou du protocole de routage unicast. Ainsi, chaque clusterhead connaît les appartenances aux groupes multicast de chaque cluster, i.e. la répartition sur les clusters des membres des groupes multicast, et il est aussi capable de construire une table de routage unicast de niveau cluster, de type "vecteur de distance" ou de type "état de liens". Le choix entre l'approche vecteur de distance ou l'approche état de liens a des répercussions sur le processus de forwarding et sur la redondance des chemins de forwarding des données multicast. Quand un clusterhead reçoit des paquets multicast, il est capable de déterminer quels sont les clusters où les destinataires i.e. les membres multicast sont et pour chaque cluster destination, il est capable de déterminer quel est le cluster vers lequel il doit faire suivre les données multicast, i.e. le cluster "next hop".

Nous avons réalisé une étude de performance sur le protocole SAFIR grâce au simulateur à évènement discret OPNET Modeler 11.5. Cette évaluation de performance avait divers objectifs :

- Vérifier l'intérêt du clustering par rapport à une solution à plat. Pour cela, nous avons comparé l'utilisation de SAFIR couplé à ODMRP pour l'intra-cluster à celle d'ODMRP sans clustering. Les courbes A.8 et A.9 montre l'overhead de contrôle et le taux de délivrance des données quand le nombre de noeuds du réseau augmente. L'intérêt de l'utilisation du clustering pour la réduction de l'overhead de contrôle est bien prouvé. De plus, nous montrons que cela ne se fait pas au détriment du taux de délivrance de données.

- Etudier l'influence de la mobilité sur le protocole SAFIR

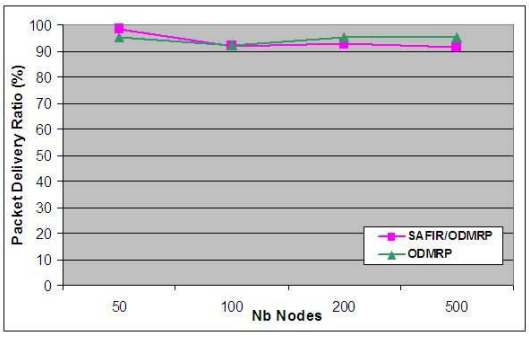

Figure A.9 Taux de délivrance des données en % en fonction du nombre de noeuds

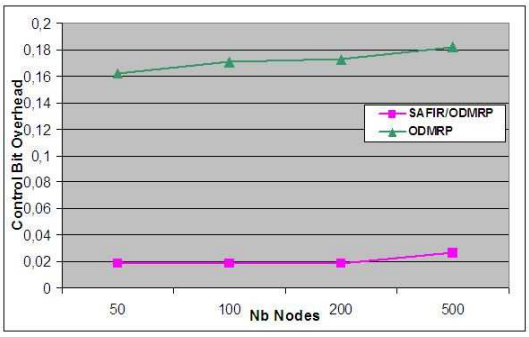

Figure A.10 Overhead de données en paquets en fonction du nombre de sources multicast par groupe

- Etudier l'influence du choix du protocole de routage multicast intra-cluster sur les performances globales du service multicast. Pour cela, on a comparé les résultats de l'association SAFIR/STAMP et SAFIR/ODMRP. On donne aussi pour référence les résultats de simulation dans le cas où ODMRP serait employé sans clustering. La figure A.10 présente les résultats obtenus pour l'overhead de données en fonction du nombre de sources multicast pour un réseau de 200 noeuds et la figure A.11 présente les résultats obtenus pour l'overhead de contrôle en fonction du nombre de groupes. En conclusion, cette étude montre que un protocole de routage qui repose sur un arbre partagé tel que STAMP permet d'obtenir de meilleurs résultats d'efficacité quant aux overheads de contrôle et de données. En effet, ce type de protocole tire mieux parti de la structure en cluster.

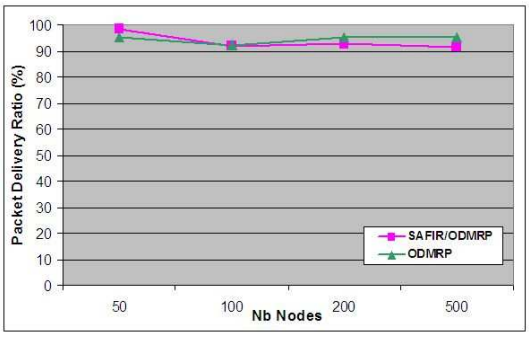

Figure A.11 Overhead de contrôle en bit en fonction du nombre de groupes multicast

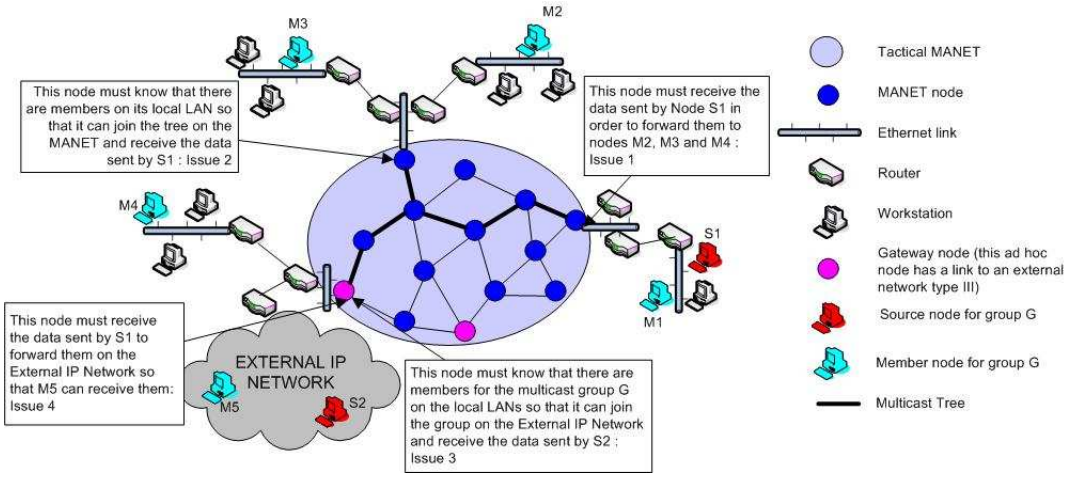

Figure A.12 Illustration des différents problème d'interconnexion à résoudre

A.5 Interopérabilité du service multicast dans le réseau Tactique

Dans la seconde partie de la thèse, nous avons présenté la structure du réseau tactique dans laquelle le réseau MANET est interconnecté avec différents types de réseaux que nous avons classé en trois types en fonction de l'architecture protocolaire qu'ils mettaient en oeuvre pour le service multicast. Nous avons montré qu'il était nécessaire de déployer dans le MANET un protocole de routage multicast spécialement conçu pour ce type de réseau. Cette utilisation souligne des questions d'interopérabilité avec les différents protocoles employés sur les autres réseaux. En effet, il est nécessaire d'implémenter des mécanismes de translation ou de proxying de façon à ce que les protocoles déployés dans le MANET inter-opèrent avec ceux déployés dans les réseaux IP connectés. Ce chapitre propose d'adresser ces problèmes d'interopérabilité en identifiant tout d'abord quels sont les problèmes à résoudre et ensuite en proposant des solutions possibles pour chacun des points soulevés. Les différentes solutions restent au stade de la proposition et devront être approfondies.

La phase d'analyse nous a permis de mettre en exergue six points nécessitant une attention pour atteindre l'interopérabilité :

- Le noeud MANET doit être capable de traiter les messages IGMP. Cela ne semble pas présenter de difficultés majeures à partir du moment où le protocole IGMP est implanté dans le noeud MANET.

- **Problème 1**. Une solution doit être proposée de façon à ce que les paquets multicast générés par une source sur un hôte local (un noeud d'un LAN local) soient acheminés jusqu'au noeud MANET local au cas où la source ne soit pas directement connectée au noeud MANET local.

- **Problème 2**. Une solution doit être proposée de façon à ce qu'un noeud MANET local connaisse les appartenances aux groupes multicast des noeuds de son LAN local dans le cas où il ne pourrait pas recevoir les messages IGMP émis par les noeuds membres.

- **Problème 3**. Une solution doit être proposée de façon à ce qu'un noeud gateway connaisse les appartenances aux groupes de tous les noeuds du réseau Tactique (les hôtes des LANs locaux ou des segments Ethernet). Dans ce point, les problèmes relatifs à la détection de gateway (**problème 5**), à la duplication de gateway (**problème 6**) et à la détection de paquets dupliqués devront être considérés.

- **Problème 4**. Une solution doit être proposée de façon à ce qu'un noeud gateway reçoive le trafic multicast généré par les hôtes des LANs locaux ou des segments Ethernet. Pour cela on peut considérer deux approches, soit le noeud gateway connaît les appartenances aux groupes multicast du coté du Réseau IP Externe auquel il est connecté, soit le noeud gateway reçoit par défaut tout le trafic généré sur les réseaux LAN et Ethernet.

La figure A.12 illustre ces différents points.

Les différentes propositions de solutions pour les différents problèmes identifiés sont résumées dans les deux tableaux suivant A.2 et A.3.

Description de la solution	Applicable si le protocole multicast employé dans le LAN est	Résolution du problème numéro	
		1	2
Le noeud MANET local est configuré comme RP	PIM-SM	**X**	**X**
l'interface Ad hoc est configuré comme receveur "wild-card" pour toutes les sources externes	Tous	**X**	
Le noeud MANET agit en tant que demandeur dans le protocole DWR	Tous		**X**

Table A.2 Solution proposées pour les problèmes 1 et 2

A.6 Qu'en est-il de la scalabilité du protocole de routage unicast?

Au cours de cette thèse, nous avons étudié le service multicast dans l'environnement tactique. Nous avons défini dans le chapitre 2 l'architecture protocolaire à mettre en place en nous attachant plus particulièrement au réseau MANET tactique. Ainsi, nous avons défini deux protocoles STAMP et SAFIR responsables des communications multicast dans le MANET. Dans la précédente partie, nous nous sommes concentrés sur les contraintes d'interopérabilité dues à l'architecture protocolaire choisie. Les protocoles que nous avons conçus pour le MANET tactique reposent sur un réseau clustérisé qui permet de répondre à la contrainte de passage à l'échelle et sur un protocole de routage unicast sous jacent. Ainsi, il convient de déterminer s'il existe dans la littérature un protocole de routage unicast qui est adapté à cette structure clustérisé.

Nous avons choisit de nous concentrer sur le protocole de routage unicast OLSR (Optimized Link State Protocol) conçu spécialement pour les réseaux MANET. Ce protocole est standardisé par l'IETF et est largement utilisé dans les déploiements de réseaux MANET. Quelques solutions ont été proposées pour augmenter la caractéristique de passage à l'échelle d'OLSR. Parmi celles-ci, seulement quelques unes (HOLSR, C-OLSR ou OLSR Tree) se fondent sur un réseau clustérisé. Ces solutions proposent d'appliquer les mêmes mécanismes qu'OLSR au niveau cluster. Pour cela ils proposent que les clusterheads échangent des super messages Hello et TC, ce qui peut générer un overhead de contrôle important. Ainsi, nous proposons une nouvelle solution d'adaptation du protocole OLSR pour des réseaux clustérisé où le protocole OLSR "normal" est appliqué dans chaque cluster mais où les communications inter-cluster ne reposent pas sur une super version d'OSLR appliquée au niveau des clusters.

Nous proposons un nouveau message, le message TC_Cluster qui est envoyé par chaque clusterhead sur tout le réseau. Grâce aux informations contenues dans ce méssage, i.e. la liste des noeuds de son cluster, n'importe quel noeud du réseau est capable de déterminer le next hop vers le clusterhead dont dépend le noeud destinataire quand la destination est située dans un cluster différent. Ces messages TC_Cluster sont diffusés sur le réseau grâce au mécanisme de flooding optimisé d'OLSR.

Nous avons évalué les performances de notre protocole grâce à une analyse théorique de l'overhead de contrôle ainsi qu'une simulation. L'évaluation de performance se concentre sur l'overhead de contrôle généré par notre protocole comparé à celui d'autres solutions telles que Fisheye OLSR et C-OLSR. d'un point de vue théorique, nous avons montré que la borne supérieure de l'overhead de contrôle de notre solution comparée à celle de Fisheye OLSR est supérieure, comme illustré dans la figure A.13. Cependant, ce résultat provient principalement de la borne supérieure sur le nombre de cluster qui est donné pour des rayons de cluster de 1 ce qui est largement sur-estimé quand le rayon augmente. Ainsi quand l'expression théorique du nombre de cluster est remplacée par une valeur simulée, la borne pour l'overhead de noter solution devient inférieure à celle de Fisheye OLSR.

D'un point de vue de la simulation, nous avons implanté les protocoles Fisheye OLSR et C-OLSR sur le simulateur Scilab. Cette simulation ne permet pas de tester les performances du protocole de bout en bout telle que le taux de délivrance des données ou la robustesse vis-à-vis de la mobilité. En effet, ces premières simulation sont une première étape et ont pour but de vérifier l'intérêt de notre solution comparée à ses concurrentes d'un point de vue de l'overhead de contrôle. Nous montrons que notre approche permet de réduire significativement l'overhead de contrôle quand le nombre de noeud du réseau augmente (cf. figureA.14).

Description de la solution	Applicable si le protocole multicast employé dans le MANET est	Résolution du problème numéro			
		3	4	5	6
Le noeud Gateway connaît les appartenances aux groupes des noeuds du MANET grâce aux messages "core_announcements"	STAMP	**X**			
Le clusterhead envoi au noeud gateway les appartenances aux groupes des noeuds du MANET à chaque fois qu'il détecte une modification.	SAFIR	**X**			
Le noeud gateway demande périodiquement au clusterhead les appartenances aux groupes des noeuds du MANET	SAFIR	**X**			
Le noeud gateway joint tous les groupes multicast pour lesquels il reçoit des messages "core_announcements". Si un noeud qui reçoit des données multicast de son réseau LAN ne connaît pas de noeud core pour le groupe, il envoi en unicast les données vers la gateway.	STAMP		**X**		
Le noeud gateway envoi un message join_all_multicast_groups à son clusterhead signifiant qu'il est membre pour tous les groupes multicast.	SAFIR		**X**		
Le noeud gateway envoi des messages d'information périodiques	Tous			**X**	
Le noeud gateway met un drapeau dans ses messages Hello pour indiquer qu'il est gateway.	Tous et AODV pour l'unicast			**X**	
Le noeud gateway envoi des messages HNA dans le réseau MANET	Tous et OLSR pour l'unicast			**X**	
Les messages TC sont modifiés pour inclure l'adresse de la gateway	Tous et OLSR pour l'unicast			**X**	
l'identifiant du noeud ou l'adresse est utilisé pour choisir la meilleure gateway	Tous				**X**

Table A.3 Solutions proposées pour les problèmes 3 à 6

A.7 Conclusion et Perspectives

A.7.1 Conclusion

La Transformation des réseaux militaires qui s'opère actuellement adopte le concept de réseau MANET en tant que composant central de l'environnement réseau tactique. Dans le futur, le réseau tactique devra pouvoir être composé de noeuds mobiles, auto-organisé, auto-configuré, qui ne reposent sur aucune infrastructure. Ainsi, le concept de réseau MANET apparaît comme la solution candidate idéale pour supporter les réseaux de communications tactiques dynamiques et mobiles.

Depuis les années 80, les Réseaux Mobile Ad hoc ont connu une attention substantielle. Ils sont souvent considérés comme la solution de réseaux d'accès sans fil pour connecter des utilisateurs mobiles à une infrastructure fixe. Leur utilisation dans le contexte militaire tactique est différente de cet emploi commercial. Alors que dans l'utilisation commerciale des réseaux mobiles ad hoc, le réseau sans fil est vu comme une extension d'un réseau IP filaire i.e. fonctionnant comme un réseau d'accès, son utilisation dans la structure du réseau tactique le place comme un réseau de transit transportant le trafic entrant, puis sortant (et non généré ou consommé par des noeuds MANET).

Le premier chapitre de ce manuscrit présente l'architecture du réseau tactique et le rôle du MANET tactique comme réseau de transit au sein de cette architecture. The réseau MANET tactique est composé d'une variété de réseaux hétérogènes tels que des LANs, des réseaux satellites, de réseaux commerciaux qui sont interconnectés au travers des Noeuds de Communications Tactiques. Ces Noeuds de Communications Tactiques intègrent des équipements radio qui forment ensemble un réseau radio hautement dynamique, le MANET tactique. Ainsi, le MANET tactique est un réseau de transit qui doit interconnecter différents

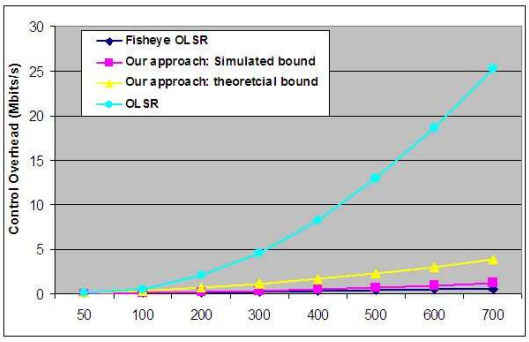

Figure A.13 Comparison des overheads des messages TC de Fisheye OLSR et de notre protocole en fonction du nombre de noeuds

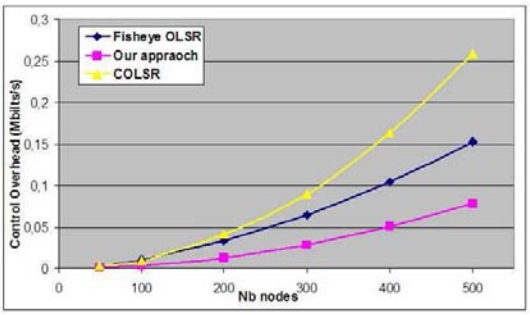

Figure A.14 Comparaison des overheads de contrôle de Fisheye OLSR, C-OLSR et de notre solution

réseaux (LANs, réseaux commerciaux Internet ...). Le contexte d'emploi du MANET tactique engendre des défis à relever en plus de ceux inhérents à ce type de réseau. On peut par exemple citer le passage à l'échelle (les réseaux tactiques peuvent être composés de plusieurs centaines de noeuds), l'importance des communications multicast et l'interopérabilité avec les réseaux filaires. Au cours de cette thèse, nous nous sommes donc efforcés de définir comment des communications multicast peuvent être mise en place entre des acteurs dispersés sur différents types de réseaux interconnectés grâce au réseau MANET tactique.

Dans le second chapitre, nous étudions l'architecture protocolaire qui doit être mise en place pour fournir un service multicast dans le réseau tactique. Nous attachons une attention particulière à la façon dont le service multicast doit être implémenté dans le MANET tactique, sachant que celui-ci est interconnecté avec différents types de réseaux qui implémentent des protocoles multicast différents. Trois solutions qui dépendent du degré d'adaptation du service multicast à l'environnement MANET ont été considérées, allant d'un noeud MANET n'implémentant aucun protocole multicast à un noeud MANET implémentant un protocole défini spécialement pour le MANET, avec un protocole défini pour les réseaux filaires comme solution intermédiaire. Nous avons conclu que la conception et l'implémentation d'un protocole spécifique MANET est la meilleure approche pour fournir un service multicast efficace dans l'environnement MANET tactique. Ce choix souligne donc un besoin pour un protocole de routage spécifique et dédié dans le MANET tactique et pose des problèmes d'interopérabilité, signifiant donc que des solutions d'interconnexion ou de proxying doivent être mise en place à l'interface avec les réseaux IP. Le passage à l'échelle étant une contrainte majeure de cet environnement, des stratégies permettant de fournir une capacité de passage à l'échelle sont étudiées et comparées. Nous arrivons à la conclusion qu'une approche avec un protocole de clustering, qui permet de rassembles les noeuds en groupes, présente les caractéristiques les plus prometteuses pour atteindre l'objectif de passage à l'échelle du MANET tactique. Ainsi nous distinguons deux niveaux de communications multicast, les communications intra-cluster quand les membres du groupe multicast sont situés au sein d'un même cluster et les communications inter-cluster quand les membres d'un groupe multicast sont dispersés dans différents clusters. Cela souligne ainsi un besoin pour un protocole de routage multicast intra-cluster qui est responsable des flux intra-cluster et pour un protocole de routage multicast inter-cluster qui est responsable des flux multicast entre les clusters.

Le troisième chapitre se concentre sur le problème de routage multicast intra-cluster. Nous présentons une nouvelle vision de l'état de l'art des protocoles de routage multicast pour les MANET pris prend l'objectif de design (robustesse, efficacité, économie d'énergie...) comme critère plutôt qu'une caractéristique du protocole telle que la topologie de la structure ou le schéma d'acquisition des routes. Cette revue permet de mettre en exergue un manque de protocoles permettant de fournir des hautes garanties de délivrance des

données tout en minimisant l'overhead, c'est-à-dire robuste ET efficace. Par conséquent, nous proposons un nouveau protocole appelé STAMP pour "Shared Tree Ad hoc Multicast Protocol" comme une alternative aux protocoles multicast existant en combinant dans le même protocole les caractéristiques d'efficacité et de robustesse. L'exigence d'efficacité est remplie grâce à une structure d'arbre partagée maintenu par une approche "hard-state" et où l'initiative de la construction de la structure partagée est donnée aux membres du groupe. Pour la robustesse, STAMP tire avantage de la capacité de diffusion du medium pour introduire de la redondance dans la structure de donnée sans augmenter le surcharge de données. Grâce à une étude de performance dans laquelle nous comparons STAMP au protocole ODMRP, l'un des plus reconnus des protocoles basés sur un mesh, nous démontrons que les objectifs de robustesse et d'efficacité sont réalisés. Dans les scénarios où les protocoles basés sur un arbre sont censés échouer en terme de taux de délivrance des données, quand la mobilité augmente par exemple, STAMP atteint des taux de délivrance des données comparable à ceux des protocoles fondés sur un mesh. d'autant plus que ces hautes garanties de délivrance de données sont atteintes avec une haute efficacité et non au prix d'une surcharge importante de messages de données et de contrôle comme dans les protocoles basés sur un mesh.

Le quatrième chapitre présente le protocole SAFIR pour ScAlable structure-Free Inter-cluster Multicast Routing comme une solution au problème de routage multicast inter-cluster. SAFIR a la responsabilité de gérer les communications multicast entre les clusters et suppose qu'un protocole de routage multicast intra-cluster tel que STAMP est déployé dans chaque cluster. Ainsi, l'objectif de ce protocole est de définir comment un datagramme multicast pour un groupe donnée sera acheminé de cluster à cluster jusqu'à atteindre les clusters où les membres du groupe multicast se trouvent. Notre protocole est optimisé en termes d'efficacité (surcharge de donnée et de contrôle). En effet, il tire partie de la signalisation envoyée par les autres services déployés sur le MANET tels que le routage unicast, ou le protocole de clustering pour envoyer les informations nécessaires à son fonctionnement propre. De plus, contrairement aux autres protocoles existants, SAFIR ne nécessite pas l'envoi de message join/reply/leave pour construire une structure multicast sur la topologie formée par les clusters. Dans ce chapitre, nous présentons également la façon dont les deux niveaux de protocole de routage multicast doivent interagir pour fournir des communications de bout en bout transparente dans le MANET tactique. L'étude d'évaluation de performances que nous avons effectuée sur SAFIR confirme que celui-ci satisfait la contrainte de scalabilité et que l'association SAFIR/STAMP présente des résultats intéressants en comparaison de l'association SAFIR/ODMRP ou d'ODMRP sans clustering.

Dans le cinquième chapitre, nous nous intéressons aux problèmes d'interopérabilité entre les protocoles de routage multicast définis pour le MANET tactique, c'est-à-dire STAMP et SAFIR et les protocoles multicast qui peuvent être déployés dans les Réseaux IP Externes ou les LAN locaux. Nous avons étudié les répercussions sur le service multicast de la répartition des différents acteurs sur les différents types de réseaux IP. Cela nous a donc amené à identifier plusieurs points qui doivent être adressés de façon à ce que les différents protocoles puissent interagir et fournir un service multicast de bout en bout, transparent pour les utilisateurs quelque soit leur positionnement dans le réseau.

Au cours de cette thèse nous nous sommes principalement concentrés sur les problématiques liées au routage multicast dans un environnement MANET tactique. Les protocoles que nous avons définis repose sur un protocole de routage unicast sous-jacent. Ainsi dans uen dernière partie, nous nous sommes intéressés aux protocoles de routage unicast résistant au passage à l'échelle. Apres une revue de l'état de l'art concernant ce sujet, il est apparut que cette problématique restait encore ouverte. Ainsi, dans ce sixième chapitre, nous présentons un protocole de routage unicast supportant le passage à l'échelle pour les réseaux MANET, qui est basé et qui améliore le protocole de routage unicast proactif le plus connu et le plus déployé, OLSR, afin de lui permettre de supporter le passage à l'échelle. Le protocole que nous avons défini est différents des autres protocoles partageant le même objectif que le notre car nous avons choisit de ne pas appliquer une "super" version d'OLSR sur la topologie formée par les clusters. En effet, nous pensons qu'une telle approche génère une surcharge de contrôle trop importante. Ainsi, nous avons défini un nouveau message qui est envoyé par chaque clusterhead et qui contient la liste des noeuds appartenant à son cluster. Ce message est utilisé pour assurer le routage inter-cluster tandis que le routage intra-cluster est assuré par le déploiement du protocole OLSR natif dans chaque cluster. Des analyses théoriques et par simulation de la surcharge de messages de contrôle en comparaison avec les autres solutions proposant des améliorations d'OLSR et se basant ou non sur du clustering permettent de montrer que notre solution permet de réduire cette surcharge de façon significative quand le nombre de noeuds du réseau augmente.

A.7.2 Perspectives

Nous donnons dans cette partie quelques idées concernant les possibles directions à explorer pour des futurs travaux sur les sujets abordés au cours de cette thèse:

- Dans STAMP, l'acheminement des données sur la structure multicast est réalisé sur l'arbre comme sur un mesh. Une amélioration de cette solution pourrait être d'adapter cet acheminement en fonction des conditions du réseau telles que la mobilité, la charge de trafic... Ainsi, l'acheminement des données pourrait passer d'une solution orientée-mesh quand les conditions sont désavantageuses (importante mobilité, par exemple), à une solution orientée-arbre plus traditionnelle quand les conditions le permettent. Cela permettrait de réduire encore plus la surcharge de données. De plus, si la couche MAC est adaptée, cela permettrait d'optimiser l'utilisation de la bande passante.

- En ce qui concerne l'évaluation de performance, plusieurs travaux peuvent être réalisés. Tout d'abord, un modèle de simulation de l'architecture globale du système pourrait être développé pour étudier les performances de différentes solutions proposées pour l'interconnexion. Ensuite, pour aller plus loin dans l'évaluation du protocole d'amélioration du protocole OLSR présentés dans le dernier chapitre, un modèle de simulation basé sur un simulateur à événement discret pourrait être développé afin d'évaluer des performances de bout en bout telles que le taux de délivrance des données ou de délai d'acheminement de bout en bout. Enfin, un modèle de simulation qui intègre tous les protocoles proposés (STAMP, SAFIR, l'amélioration d'OSLR, les solutions d'interopérabilité) peut aussi être envisagé. Pour le moment, SAFIR et STAMP ont d'hors et déjà été intégrés.

- Le travail d'évaluation de performance que nous avons effectué au cours de cette thèse avait pour but d'évaluer les protocoles dans un contexte général, avec un placement aléatoire des noeuds, une mobilité aléatoire, ou le modèle du 802.11e comme couche MAC par exemple. Cela représenté la première étape du processus d'évaluation de performance. L'étape suivante sera d'évaluer les protocoles dans un contexte plus spécifique. Ainsi, la couche MAC sera remplacée par une couche MAC plus représentative du système cible sur lequel les différents protocoles doivent être déployés. De la même façon, les scénarios choisit pour l'évaluation seront plus représentatifs de scénarios opérationnels tant que le plan de la mobilité, que sur le plan du nombre de noeuds, du positionnement des noeuds, de la charge de trafic ou de la répartition des acteurs multicast. Enfin, les protocoles seront évalués avec des profils de trafic représentatif des applications réelles plutôt que du trafic CBR. Par exemple, le module "System-In-The-Loop" d'OPNET pourra être utilisé pour injecter du trafic réel telle qu'un flux vidéo dans la simulation.

- Enfin, la dernière étape sera d'implémenter les protocoles proposés dans le produit final pour évaluer les performances des protocoles dans des scenarios de déploiement opérationnels. Cette étape sera réalisée pour le protocole STAMP dans l'année à venir.

En plus de ces perspectives qui se rapportent directement à l'amélioration du travail présenté dans cette thèse, nous pouvons envisager des directions de recherche plus générale. En effet, maintenant que le protocole STAMP qui permet de répondre aux exigences de robustesse, d'efficacité et d'économie d'énergie a été défini, il serait intéressant de s'en servir de base pour adresser les autres exigences qu'un protocole de routage multicast doit satisfaire dans un environnement MANET tactique, à savoir la QoS, la sécurité et la fiabilité. Ainsi, les protocoles STAMP et SAFIR, définis au cours de cette thèse pourraient être enrichis par des mécanismes additionnels permettant d'adresser la liste d'exigences définies au chapitre 2. La prochaine étape serait donc d'incorporer des possibilités de routage QoS, des mécanismes de sécurité tels qu'un schéma de gestion des clés et des protocoles de fiabilisation. Pour chacun de ces services, des études sur l'interopérabilité devront être réalisées de façon à fournir un service multicast de bout en bout transparent pour l'utilisateur. Enfin, la version IPv6 est attendue comme étant le futur de l'Internet Protocol qui est aujourd'hui la version IPv4. IPv6 offre des solutions aux limitations d'IPv4 qui n'avait pas été conçu pour le type de réseau que l'Internet est devenu aujourd'hui. Ainsi, dans le contexte du multicast pour les réseaux MANET tactiques, des recherches devront être effectuées de façon à évaluer l'impact de la migration vers IPv6. Ce travail inclurait l'évaluation de STAMP et SAFIR de façon à identifier les changements qui devraient être effectués pour que ces protocoles s'intègrent dans un environnement IPv6.

Publications

International Conference

Lucile Canourgues, Jerome Lephay, Laurent Soyer, Andre-Luc Beylot, "An efficient Scalable structure-Free Intercluster multicast Routing (SAFIR) protocol for large tactical Mobile Ad hoc Networks", In IEEE Military Communications Conference (MILCOM 2007), Orlando, USA, 29/10/2007-31/10/2007, IEEE, October 2007 (to be published)

Lucile Canourgues, Jerome Lephay, Laurent Soyer, Andre-Luc Beylot. "STAMP : Shared-Tree Ad hoc Multicast Protocol". In IEEE Military Communications Conference (MILCOM 2006), Washington, 22/10/2006-25/10/2006, IEEE, p. 174-180, Octobrer 2006.

Lucile Canourgues, Jerome Lephay, Laurent Soyer, Andre-Luc Beylot. "A Scalable Adaptation of the OLSR Protocol for Large Clustered Mobile Ad hoc Networks ". In 7th IFIP Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net 2008), Palma de mallorca, Spain , 23-27 June 2008, Paper accepted

Conference articles without published proceedings

Lucile Canourgues, "Shared-Tree Ad hoc Multicast Protocol (STAMP) : un protocole de multicast efficace fonde sur la capacite de diffusion du medium sans fil" In Colloque des Doctorants - EDIT'07, Toulouse, 24/05/2007-25/05/2007.

Patent

Lucile Canourgues, Patent deposed on the 16th April 2007, "Multicast Routing for a clustered mobile ad hoc network", for Rockwell Collins France, Application Number EP07106262

List of Acronyms

A

ABAM	On-Demand Associativity-Based Multicast
ACK / NACK	ACKnowledgment / Negative ACKnowledgment
ADMR	Adaptive Demand-driven Multicast Routing
AMRIS	Ad hoc Multicast Routing protocol Utilizing Increasing Id numbers
AMRoute	Ad hoc Multicast Routing
AODV	Ad hoc On-demand Distance Vector
AP	Access Point
ARQ	Automated Repeat Request
AS	Autonomous System
ASTM	Adaptive Shared Tree Multicast
ATH	At The Halt

B

BEMR	Bandwidth-efficient multicast routing
BGMP	Border Gateway Multicast Protocol
BGP	Border Gateway Protocol

C

CAMP	Core-Assisted Mesh Protocol
CBMRP	Cluster-Based Multi-Source Multicast Routing Protocol
CBO	Control Bit Overhead
CBR	Constant Bit Rate
CBT	Core Based Tree
CNR	Combat Net Radio
C-OLSR	Clustered OSLR
COTS	Commercial Off The Shelf
CQMP	Consolidated Query-packet Multicast. Protocol

D

DARPA	Defense Advanced Research Projects Agency
DCMP	Dymanic Core Multicast Protocol
DDM	Differential Destination Multicast
DM	Dense Mode
DoD	Department of Defense
DPO	Data Packet Overhead
DPUMA	Differential Protocol for Unified Multicasting through Announcements
DR	Designated Router
DSDV	Destination-Sequenced Distance-Vector
DSR	Dynamic Source Routing
DVMRP	Distance Vector Multicast Routing Protocol
DWR	Domain Wide Multicast Group Membership Reports

E

ETE	End-To-End
EXPRESS	Explicit Requested Single Source

F

FGMP-RA	Forwarding Group Multicast Protocol - Receiver Advertisement
FGMP-SA	Forwarding Group Multicast Protocol - Sender Advertisement
FHMR	First Hop Multicast Router

G

GBMP	Gateway Based Multicast Protocol
GIG	Global Information Grid
GKMP	Group Key Management Protocol
GPS	Global Positioning System

H

HDDM	Hierarchical DDM
HIM-TORA	Hierarchical Multicast - Temporally-Ordered Routing Algorithm
HMP	Hierarchical Multicast Protocol
HNA	Host and Network Association
H-OLSR	Hierarchical OLSR

I

IETF	Internet Engineering Task Force
IGMP	Internet Group Management Protocol
IP	Internet Protocol

K

KEK	Key Encryption Key

L

LAM	Lightweight Adaptive Multicast
LAN	Local Area Network
LANMAR	Landmark Ad Hoc Routing Protocol
LAS	Local Area Subsystem
LMT	Lifetime-aware Multicast Tree
L-REMIT	Lifetime-Refining Energy efficient of Multicast Trees
LSA	Link State Advertisement

M

MAC	Medium Access Control
MANET	Mobile Ad hoc NETwork
MANSI	Multicast for Ad Hoc Networks with Swarm Intelligence
MAODV	Multicast Ad-hoc On demand Distance Vector
MASC	Multicast Address-Set Claim
Mbone	Multicast Backbone
MBR	Multicast Border Router
MHMR	Mobility-based Hybrid Multicast Routing
MLANMAR	Multicast Landmark Ad Hoc Routing Protocol
MOLSR	Multicast Optimized Link State Protocol

MOSPF	Multicast Open Shortest Path First
MPR	Multi-Point Relay
MRDC	Multicast Routing protocol with Dynamic Core
MSDP	Multicast Source Discovery Protocol
MSTP	Multicast Shared Tree Protocol
MTP	Multicast Transfer Protocol
MZRP	Multicast Zone Routing Protocol

N

NBD	Network-Based Defense
NCW	Network Centric Warfare
NEC	Network-Enabled Capabilities
NSMP	Neighbor Supporting ad hoc Multicast routing Protocol
NTIC	New Technologies of Information and Communications

O

O/H	OverHead
ODMRP	On Demand Multicast Routing Protocol
ODMRP-MPR	ODMRP-Multi Point Relay
ODMRP-PDA	ODMRP-Passive Data Acknowledgement
OLSR	Optimized Link State Routing
OSPF	Open Shortest Path First

P

PAST-DM	Progressively Adapted Sub-Tree in Dynamic Mesh
PDR	Packet Delivery Ratio
PhD	Doctor of Philosophy
PIM	Protocol Independent Multicast
PMBR	PIM Multicast Border Router
POMA	Prioritized Overlay Multicast Ad hoc
PRNET	Packet Radio NETwork
PUMA	Protocol for Unified Multicasting through Announcements

Q

QMR	QoS for Multicast Routing
QoS	Quality of Service

R

RFC	Request For Comment
RIP	Routing Information Protocol
ROMANT	RObust Multicasting in Ad hoc Networks using Trees
RP	Rendezvous Point
RPM	Reverse Path Multicasting

S

SA	Situational Awareness
SA	Source-Active
SAFIR	Scalable Structure-Free Inter-cluster multicast Routing
SM	Sparse Mode
SM	Simple Multicast
SMMRP	Scalable Multi-source Multicast Routing Protocol
SRMP	Source Routing-based Multicast Protocol
STAMP	Shared-Tree Ad hoc Multicast Protocol

SURAN	SUrvivable RAdio Network

T

TC	Topology Control
TCN	Tactical Communication Nodes
TCP	Transmission Control Protocol
TEK	Transmission Encryption Key
TPO	Total Packet Overhead
TRPB	Truncated Reverse Path Broadcasting
TTL	Time To Live

U

UAVs	Unmanned Aerial Vehicles
UDP	User Datagram Protocol
US	United-State

W

WAS	Wide Area System
WCMRP	Weight-based Clustering Multicast Protocol
WG	Working Group
WLAN	Wireless Local Area Network

Resume

La Transformation qui s'opère depuis quelques années dans les réseaux militaires place le réseau MANET comme une composante principale du domaine tactique. En effet, un réseau MANET permet de mettre en oeuvre des noeuds de communication de grande mobilité, de grande réactivité et qui se déploient rapidement. De nombreuses applications militaires temps réel, telles que le "Situational Awareness", reposent sur des communications de groupes et nécessitent donc la mise en place d'un service multicast dans l'environnement tactique où le réseau MANET est utilisé comme réseau de transit. L'objectif de cette thèse est d'étudier la mise en place d'un service multicast optimum dans cet environnement tactique MANET. Nous nous sommes tout d'abord attachés à définir l'architecture protocolaire multicast à mettre en oeuvre au sein du réseau tactique en attachant une attention particulière au réseau MANET. Ce réseau est interconnecté avec différents types de réseaux reposant sur une technologie IP et employant des protocoles multicast potentiellement hétérogènes. Le réseau MANET tactique est supposé pouvoir mettre en oeuvre plusieurs centaines de noeuds, ce qui implique que la contrainte de passage à l'échelle est déterminante dans le choix de l'architecture protocolaire du service multicast. Le concept de clustering présentant de bonnes caractéristiques de passage à l'échelle, nous avons donc considéré le réseau MANET comme un réseau clusterisé. Nous avons pu alors définir deux protocoles de routage multicast adaptés aux réseaux MANET: tout d'abord STAMP qui est en charge des communications multicast à l'intérieur de chaque cluster et ensuite SAFIR qui se charge des flux multicast entre les clusters. Ces deux protocoles, qui peuvent être implémentés indépendamment, agissent de concert pour fournir un service multicast performant et supportant le passage à l'échelle dans le réseau MANET tactique. Ensuite, nous avons étudié l'interopérabilité de ces protocoles multicast employés dans le MANET avec ceux employés sur les réseaux hétérogènes interconnectés avec celui-ci de façon à garantir un service multicast de bout-en-bout transparent pour les utilisateurs. Enfin, les protocoles multicast proposés au cours de cette thèse étant directement liés aux protocoles de routage unicast sous-jacents, nous avons proposé, dans une dernière partie, une solution de routage unicast reposant sur OLSR et tolérant le passage à l'échelle.

Abstract

The current Transformation of the military networks adopts the MANET as a main component of the tactical domain. Indeed, a MANET is the right solution to enable highly mobile, highly reactive and quickly deployable tactical networks. Many applications such as the Situational Awareness rely on group communications, underlying the need for a multicast service within the tactical environment where the MANET is employed as a transit network. The purpose of this thesis is to study the setting up of an optimal multicast service within this tactical environment. We firstly focus on defining the protocol architecture to carry out within the tactical network paying particular attention to the MANET. This network is interconnected with different types of networks based on IP technologies and implementing potentially heterogeneous multicast protocols. The tactical MANET is supposed to be made of several hundred of mobile nodes, which implies that the scalability is crucial in the multicast protocol architecture choice. Since the concept of clustering proposes interesting scalability features, we consider that the MANET is a clustered network. Thereby, we define two multicast routing protocols adapted to the MANET: firstly STAMP that is in charge of the multicast communications within each cluster and secondly SAFIR that handles multicast flows between the clusters. These two protocols that can be implemented independently, act in concert to provide an efficient and scalable multicast service for the tactical MANET. Then, we study the interoperability of these multicast protocols employed within the MANET with those employed in the heterogeneous networks that it is interconnected with in order to guarantee end-to-end seamless multicast services to users. Finally, since the multicast protocols proposed in this thesis rely on underlying unicast routing protocols, we propose, in the last chapter, a scalable unicast routing protocol based on OLSR.