Universidade de Aveiro Departamento de Electrónica, Telecomunicações e Informática
2014

Programa de Doutoramento em Informática das Universidades do Minho, Aveiro e do Porto

Lucas Guardalben

Comunicação entre Nós para Gestão Autonómica e Distribuída

Communication between Nodes for Autonomic and Distributed Management

Universidade de Aveiro 2014
Departamento de Electrónica, Telecomunicações e Informática
Programa de Doutoramento em Informática das Universidades do Minho, Aveiro e do Porto

Lucas Guardalben

Comunicação entre Nós para Gestão Autonómica e Distribuída

Communication between Nodes for Autonomic and Distributed Management

Tese apresentada às Universidades de Minho, Aveiro e Porto para cumprimento dos requisitos necessários à obtenção do grau de Doutor no âmbito do doutoramento conjunto MAP-i, realizada sobre a orientação científica da Doutora Susana Isabel Barreto de Miranda Sargento, Professora Auxiliar do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro, e coorientação do Doutor Paulo Jorge Salvador Serra Ferreira, Professor Auxiliar do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro.

O trabalho desenvolvido no decorrer desta tese foi financiado pela Bolsa de Doutoramento SFRH/BD/62511/2009, financiada pela Fundação para a Ciência e Tecnologia (FCT).

o júri / the jury

presidente / president	**Professor Doutor Helmuth Robert Malonek** Professor Catedrático da Universidade de Aveiro
vogais / examiners committee	**Professor Doutor Edmundo Heitor da Silva Monteiro** Professor Catedrático da Universidade de Coimbra
	Professor Doutor Rui Carlos Mendes Oliveira Professor Associado da Universidade do Minho
	Professora Doutora Teresa Maria Sá Ferreira Vazão Vasques Professora Associado do Instituto Superior Técnico de Lisboa
	Professora Doutora Susana Isabel Barreto de Miranda Sargento Professora Auxiliar da Universidade de Aveiro (Orientadora)
	Doutor Rui Lopes Campos Investigador Doutorado do INESCTEC do Porto

"The man who moves a mountain begins by
carrying away small stones".
by Confucius.

agradecimentos / acknowledgments

Gostaria primeiramente de agradecer a Deus e à minha querida tia Atília Guardalben por estarem sempre presentes em minhas orações, como também a todos os membros da minha família; em especial aos meus pais, Aleri Luiz Guardalben e Ivete Ana Guardalben e meus irmãos Cássio Fernando Guardalben e Laís Guardalben, por todo apoio e palavras confortantes nos momentos certos.
Aos meus orientadores: Professora Susana Sargento e Professor Paulo Salvador, pelas discussões, ensinamentos e orientações necessárias durante o início, meio e o fim deste trabalho. Aos professores das Universidades: Aveiro, Porto e Minho pela contribuição no meu aprendizado, fornecendo a base necessária para um bom trabalho. À FCT (Fundação de Ciência e Tecnologia) pelo apoio financeiro, o qual contribuiu de forma preponderante para o avanço e conclusão do meu doutoramento.
Agradeço em especial à minha esposa Marcela Vaz pela compreensão e por dividir comigo todos os momentos de alegria e dificuldade durante todo esse percurso. Agradeço aos grandes amigos que tive a chance de conhecer aqui em Portugal por ajudarem a concretizar este sonho. Aos amigos do grupo de investigação Network Architectures and Protocols (NAP) por toda ajuda e momentos de descontração, contribuindo fortemente para meu crescimento profissional e pessoal.
Ao Instituto de Telecomunicações (IT-AVEIRO) por disponibilizar toda infraestrutura necessária para concluir este trabalho e aos membros do juri pela atenção despendida na leitura e sugestões para melhoria do trabalho.
Por fim!
MUITO OBRIGADO a todos que direta e indiretamente contribuíram para a conclusão deste doutoramento.

Palavras-chave

Gestão Autonómica e Distribuída, Comunicação entre Nós, (Re) Organização dos Nós da Rede, Disseminação de Informações de Gestão.

Resumo

Durante a última década, protocolos como Simple Network Management Protocol (SNMP) ou Common Management Information Protocol (CMIP) foram as abordagens mais comuns para a gestão tradicional de redes. Essas abordagens têm vários problemas em termos de escalabilidade, devido às suas características de centralização. Apresentando um melhor desempenho em termos de escalabilidade, as abordagens de gestão distribuída, por sua vez, são vantajosas nesse sentido, mas também apresentam uma série de desvantagens acerca do custo elevado de comunicação, autonomia, extensibilidade, flexibilidade, robustez e cooperação entre os nós da rede. A cooperação entre os nós presentes na rede é normalmente a principal causa de sobrecarga na rede, uma vez que necessita de colectar, sincronizar e disseminar as informações de gestão para todos os nós nela presentes. Em ambientes dinâmicos, como é o caso das redes atuais e futuras, espera-se um crescimento exponencial no número de dispositivos, associado a um grau elevado de mobilidade dos mesmos na rede. Assim, o grau elevado de funções de automatização da gestão da rede é uma exigência primordial, bem como o desenvolvimento de novos mecanismos e técnicas que permitam essa comunicação de forma optimizada e eficiente. Tendo em conta a necessidade de elevada cooperação entre os elementos da rede, as abordagens atuais para a gestão autonómica permitem que o administrador possa gerir grandes áreas de forma rápida e eficiente frente a problemas inesperados, visando diminuir a complexidade da rede e o fluxo de informações de controlo nela gerados. Nas gestões autonómicas a delegação de operações da rede é suportada por um plano auto-organizado e não dependente de servidores centralizados ou externos. Com base nos tipos de gestão e desafios acima apresentados, esta Tese tem como principal objetivo propor e desenvolver um conjunto de mecanismos necessários para a criação de uma infra-estrutura de comunicação entre nós, na tentativa de satisfazer as exigências da gestão autonómica e distribuída apresentadas pelas redes de futura geração. Nesse sentido, mecanismos específicos incluindo inicialização e descoberta dos elementos da rede, troca de informação de gestão, (re) organização da rede e disseminação de dados foram elaborados e explorados em diversas condições e eventos, tais como: falhas de ligação, diferentes cargas de tráfego e exigências de rede. Para além disso, os mecanismos desenvolvidos são leves e portáveis, ou seja, podem operar em diferentes arquitecturas de hardware e contemplam todos os requisitos necessários para manter a base de comunicação eficiente entre os elementos da rede. Os resultados obtidos através de simulações e experiências reais comprovam que os mecanismos propostos apresentam um tempo de convergência menor para descoberta e troca de informação, um menor impacto na sobrecarga da rede, disseminação mais rápida da informação de gestão, aumento da estabilidade e a qualidade das ligações entre os nós e entrega eficiente de informações de dados em comparação com os mecanismos base analisados. Finalmente, todos os mecanismos desenvolvidos que fazem parte da infrastrutura de comunicação proposta foram concebidos e desenvolvidos para operar em cenários completamente descentralizados.

Keywords

Autonomic and Distributed Management, Communication Between Nodes, Network (Re) Organization, Management Information Dissemination.

Abstract

Over the last decade, the most widespread approaches for traditional management were based on the Simple Network Management Protocol (SNMP) or Common Management Information Protocol (CMIP). However, they both have several problems in terms of scalability, due to their centralization characteristics. Although the distributed management approaches exhibit better performance in terms of scalability, they still underperform regarding communication costs, autonomy, extensibility, flexibility, robustness, and cooperation between network nodes. The cooperation between network nodes normally requires excessive overheads for synchronization and dissemination of management information in the network. For emerging dynamic and large-scale networking environments, as envisioned in Next Generation Networks (NGNs), exponential growth in the number of network devices and mobile communications and application demands is expected. Thus, a high degree of management automation is an important requirement, along with new mechanisms that promote it optimally and efficiently, taking into account the need for high cooperation between the nodes. Current approaches for self and autonomic management allow the network administrator to manage large areas, performing fast reaction and efficiently facing unexpected problems. The management functionalities should be delegated to a self-organized plane operating within the network, that decrease the network complexity and the control information flow, as opposed to centralized or external servers. This Thesis aims to propose and develop a communication framework for distributed network management which integrates a set of mechanisms for initial communication, exchange of management information, network (re) organization and data dissemination, attempting to meet the autonomic and distributed management requirements posed by NGNs. The mechanisms are lightweight and portable, and they can operate in different hardware architectures and include all the requirements to maintain the basis for an efficient communication between nodes in order to ensure autonomic network management. Moreover, those mechanisms were explored in diverse network conditions and events, such as device and link errors, different traffic/network loads and requirements. The results obtained through simulation and real experimentation show that the proposed mechanisms provide a lower convergence time, smaller overhead impact in the network, faster dissemination of management information, increase stability and quality of the nodes associations, and enable the support for efficient data information delivery in comparison to the base mechanisms analyzed. Finally, all mechanisms for communication between nodes proposed in this Thesis, that support and distribute the management information and network control functionalities, were devised and developed to operate in completely decentralized scenarios.

Contents

List of Figures

List of Tables

List of Algorithms

Abbreviations

4WARD	Architecture and Design for Future Internet
AC	Autonomic Computing
ADS	Autonomic Decision System
ADUs	Application Data Units
ALBA	Autonomic Load Balancing Algorithm
AM	Autonomic Manager
AMSs	Autonomic Management Systems
ANA	Autonomic Network Architecture
ANEMA	Autonomic NEtwork MAnagement
ANs	Active Networks
AON	Application-Oriented Networking
AORTA	Autonomic Network Control and Management System
AP	Access Point
ARB	Autonomic Resource Brokers
ASA	Autonomic Service Architecture
AWARE	Coordination Action for Self-Awareness in Autonomic Systems
AUTOI	Autonomic Internet
BSNs	Bayesian Networks
BSS	Basic Service Set
BSSID	Basic Service Set Identifier
CAN	Content Addressable Network
CAPEX	Capital Expenditure
CASCADAS	Component aware for Autonomic, Situation-aware Communications and Dynamically Adaptable
CDP	Cisco Discovery Protocol

CHOP	Configuration, Healing, Optimization and Protection
CLI	Common Line Interfaces
CMIP	Common Management Information Protocol
CMIS	Common Management Information Service
CORBA	Common Object Request Broker Architecture
CLD	Critical Link Disruptor
CND	Critical Node Disruptor
CRC	Cyclic Redundancy Check
CUE	Channel Utilization Estimation
DAG	Direct Acyclic Graphs
DCF	Distributed Coordination Function
DDNS	Dynamic Domain Name System
DS	Domain System
DSR	Dynamic Source Routing
DSSS	Direct Sequence Spread Spectrum
DtD	Device-to-Device
DTNs	Delay Tolerant Networks
ESS	Extendend Service Set
ETSI	European Telecommunications Standards Institute
FAP	Fast Association Process
FCAPS	Fault, Configuration, Accounting, Performance, Security management
FHSS	Frequency Hopping Spread Spectrum
FN	Forwarding Node
FOCALE	Foundation Observation Comparison Action Learn Reason
FP7	EU Framework Programme 7
GUI	Graph User Interface
HIS	Haggle Information Space
HISK2D	HIde and SeeK Directional Discovery
IBSS	Independent Basic Service Set
IDM	Information Dissemination Management

IE	Information Element
IEEE	Institute of Electrical and Electronics Engineers
INM	In-Network Management
ISO	International Standard Organization
IT	Information Technology
ITU-T	International Telecommunication Union
LANs	Local Area Networks
LSAs	Link State Advertisements
LSR	Link State Routing
MAC	Media Access Control
MA-Fi	Mobile Ad-Hoc Wi-Fi
MANET's	Ad-Hoc Networks
MANNA	Management Architecture for Wireless Sensor Networks
MbD	Management by Delegation
MIBs	Management Information Bases
MPR	Multipoint Relay
NED	Neighbors Eyesight Direction
NGNs	Next Generation Networks
NGs	Next Generation Service
NLE	Network Leader Election
NMS	Network Management System
NMSs	Network Management Stations
NS-3	Network Simulator v.3
OFDM	Orthogonal Frequency Division Multiplexing
OLSR	Optimized Link State Routing
OPEX	Operational Expenditure
OSI-SM	International Organization for Standardization System Management
OSI-SM	OSI System Management
OSKMV	Orchestration Plane, Service Enablers Plane, Knowledge Plane, Management Plane and Virtualization Plane

PACMAN	Platform for Automated and Controlled network operations and configuration MANagement
PBM	Policy-Based Management
RMON	Remote Network MONitoring
RSSI	Received Signal Strength Indication
RTT	Round-Trip-Time
Self-X	Configuration, Healing, Optimization and Protection
SLA	Service-Level Agreement
SMIs	Management Information's
SNMP	Simple Network Management Protocol
SNR	Signal to Noise Ratio
SOA	Service-Oriented Architecture
SDNs	Software Defined Networks
SSID	Service Set IDentifier
STA	Station
STP	Spanning Tree Protocol
TMN	Telecommunication Management Network
WEP	Wired Equivalent Privacy
WMNs	Wireless Mesh Networks
WSTP	Wireless Spanning Tree Protocol
XML	Extensible Mark-up Language

Chapter 1

Introduction

This Chapter introduces the scope of the work and motivation towards the definition and evaluation of a communication framework for distributed network management. Furthermore, we emphasize the main objectives through research questions which help to understand the work performed in this Thesis. Moreover, we present the main contributions of the Thesis as the result of exploring the proposed new concepts and innovative research directions. Finally, this Chapter ends with the outline of the Thesis.

1.1 Scope of Work and Motivation

Successful communication is a pre-requisite for the progress of global society. Constantly reinforced by the development of new technologies and services, the communication is fully indispensable in all community levels. Upon the technological boom of late 90s, the communication tools have undergone considerable improvement. The progress of communication technology, coupled with immeasurable commercialization and explosion of sales, enabled the use of a large number of network accessories and communication platforms. In addition, it broke down the geographical barriers establishing the levels of globalization necessary for further commercialization. Those technological advances aim at facilitating human interactions and are constantly updated for their capabilities and intelligence. The following device diversification is responsible for the traffic growth by increasing the number of devices accessing mobile networks worldwide [Inc13]. The low cost of up-front devices and their easy integration in the mobile communication infrastructure introduced new opportunities for networking innovations. The market tendency shows that different networks will have to interact in a global network where multiple access technologies will be simultaneously available. The choice of access network scenario can depend on technical (achievable bitrate, minimum delay) or non-technical (cost) decision criterion's. Based on this decision, the network elements and terminals can vary from simplest (e.g. a sensor), to the most complex (e.g. a server or a mobile router). Moreover, the changes in the network can occur naturally, when a node arrives or departures from the network, intentionally or by malfunction.

Perhaps the most obvious implication of the importance of network management can be found through Information Technology (IT) point of view. Management of network infrastructures can reduce time and money to their Capital Expenditure (CAPEX) and Operational Expenditure (OPEX) operations, enhancing administration, controlling and provisioning their network infrastructures [Cle07]. And in this sense, the IT managers can control and maintain their IT networks by centralizing and automating all activities related to planning, tracking, monitoring, servicing, and fine-tuning their network components through management approaches as well. However, over the last decade, the most widespread approaches for network management entailed the use of centralized approaches, such as the Simple Network Management Protocol (SNMP) [Cea90] or the Common Management Information Protocol (CMIP) [Wea90]. This centralization has benefits specifically for networks of moderate size, whose configuration rarely change and whose states evolve slowly adaptations and thus do not require fast intervention by an outside system. Although distributed approaches have better performance in terms of scalability, they still bare disadvantages concerning communication costs, autonomy, extensibility, flexibility, robustness, and cooperation between network nodes [AIH97, Bea00, Cea02, Pea02]. The high level of management automation tasks is an important requirement in distributed management.

In distributed management approaches, as performed in the scope of this Thesis, each device interacts with its peers, taking decisions based on the gathered knowledge from the other devices, and performing a network of collaboration and cooperation over multi-network technologies environments. The idea is that management tasks or functions can be delegated from management stations outside the network to a self-organizing management plane inside the network. Thus, each network management process interacts with the neighborhood through peer-to-peer interaction and relying on different propagation schemes to enforce management processes that will allow a level of real-time awareness notification and management reconfiguration. This approach requires continuous interactivity between entities in order to exchange information about each entity (and therefore the network). This information will allow the network to make autonomic decisions, through collaboration between the network nodes, reacting to network changes (such as link failures, load variations, etc) and continuously optimizing the network resources. In order to realize the totally distributed management plane, a management framework with processing and communication functions is associated with each network element or device, which, in addition to collecting information, monitors and configures local parameters and communicates with peer entities in its proximity. The collection of these entities creates a thin layer of management functionality inside the network that can take advantage of distributed scenarios, thus, monitoring and control tasks end-to-end.

In the Figure 1.1, a comparison between network management approaches is depicted. In the traditional network management, the network administrator has the centralized control to perform network management decisions, interacting with the network management through a single command interface. In centralized self-management approaches, the control and decisions are subject to the control-loop in an automatic way. Therefore, most of the self-management approaches use centralized servers to control, act and disseminate the policies and rules. We argue that the external server approaches turned out to be inadequate in terms of scalability of the managed system.

In this Thesis the main focus is in the self distributed network management, with the goal to achieve scalable and low complexity management for dynamic multiple network technologies environments. The potential benefits of this decentralization include: (i) a high level of scalability of management systems, which has small execution times and low traffic overhead in large-scale managed systems; (ii) effective management of large networks; and (iii) fast reaction in response to local or global events, specially in scenarios with unstable communications. As result, this will increase the adaptability of the network to various types of faults, configuration changes, load changes, etc.

In addition, high cooperation ability between the nodes through stringent communication infrastructures is crucial for scalability of the information exchange, and minimization of the amount of control messages flowing in the network. Ultimately, an ideal management approach needs to be able to adapt itself to the constant network evolution, overcoming the limitations through a tight coordination and cooperation between the heterogeneity of net-

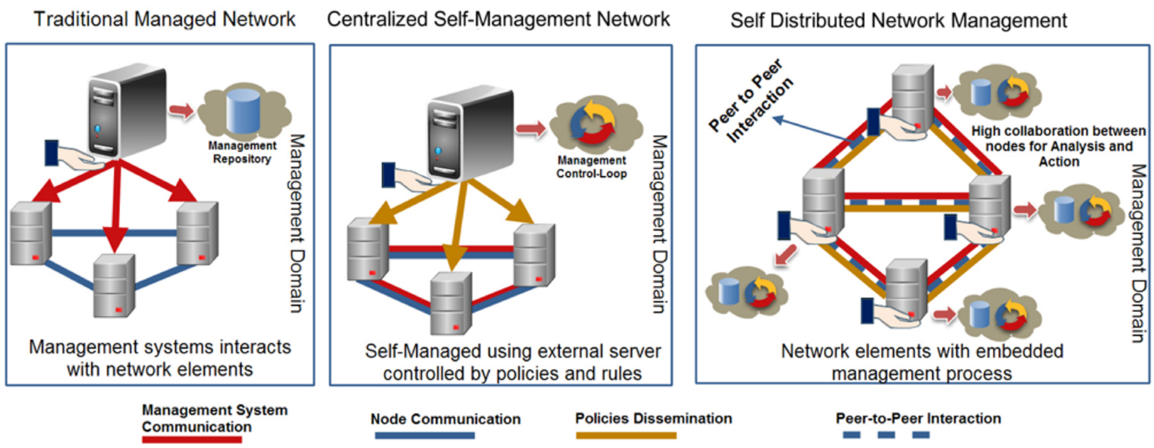

Figure 1.1: Traditional, centralized self-management and totally self distributed management

works and nodes. This explains why cooperation and delegation between nodes will be highly required. As such, the so-called Next Generation Networks (NGNs) imply novel challenges compared to those encountered in today's networks [BS13, Din10]. For instance, NGNs use flexible and efficient radio access that permits pervasive access to Internet through mobile devices. They also require real-time management of new forms of ad-hoc communications with high level of mobility, intermittent connectivity and time-varying network topology. NGNs need to integrate the variety of sub-networks towards network convergence, e.g., pocket, personal, sensor and micro sensor networks. Furthermore, they will need to overcome bandwidth limitations in access, and enable intelligent distribution of content; for this process to efficiently work, it is fundamental to use distributed or completely decentralized control, and also to enable seamless end-to-end network and service composition.

Thus, IT managers and telecom operators need to rethink the management techniques that will be used to respond quickly to this challenge with the purpose of increasing the quality of service for their customers. This is only achievable through distributed management approaches which bring inherent support for self-management capabilities, enhancing the network with inbuilt intelligence. The current trend of distributed management demands the embedding of management functionalities within the network nodes. However, this embedding requires the network nodes to have sufficient information to take management decisions in an efficient and cooperative way. Thereafter, this information needs to be exchanged, which may increase the network overhead to unbearable limits. Therefore, new lightweight approaches - reducing the use of network resources to a minimum - need to be developed in order to organize the network nodes and determine the interactions between them, enabling the optimization of the network information exchange. The term lightweight refers to the use of network minimum resources possible in order to provide a base for communication between the nodes. To overcome the aforementioned challenges, new network communication infrastructures for supporting distributed management functionalities need to be designed using clean slate approach, without inheriting the limitations of other approaches.

In this scope, we propose a communication framework for self distributed network management with the purpose of providing the proper information for management tasks and functions, residing in each network device. The management functionalities are delegated to a self-organized management plane that operates inside the network, reducing the complexity and network information flow compared to the centralized approach. In this context, we target the development of mechanisms with the purpose of: (i) bootstrapping and discovery in order to provide initial communication between the nodes; (ii) enabling the exchange of partial information between non directly connected nodes; (iii) network (re) organization to perform nodes association and, (iv) data dissemination to enhance data delivery. The aforementioned four topics are the focus of most of the proposed mechanisms within the research work performed in this Thesis. Furthermore, we explore our proposed framework in diverse conditions and events, such as device and link errors, different traffic loads and diverse network requirements in a multi-network technologies testbed. Finally, due to the scalability limitations of centralized management solutions, we propose all mechanisms to support and distribute the management information and network control functionalities in a complete decentralized way. The objectives related to the mechanisms towards achieving cooperation through communication framework for distributed network management are presented in the following Section 1.2.

1.2 Research Objectives

This Thesis aims to develop a communication framework to provide a management communication infrastructure, attempting to overcome the management requirements posed by NGNs. The mechanisms are the key to provide an efficient, scalable, and robust management support with low complexity and message overhead for large-scale, dynamic network environments. This Thesis addresses the development of mechanisms and functionalities in order to deal with the following specific objectives.

1.2.1 Research Objective: I

In which way must the nodes exchange initial information to reduce communication costs?

Nodes need to communicate to discover connected nodes, obtain their management information and roles, as well as optimize the way they communicate and operate. As referred in the state of the art, the mechanisms to be provided ought to offer low cost communication in a fully-distributed way. The information that needs to be exchanged to optimize the performance should also be specified. The proposal of low overhead impact on the communication between nodes is an important requirement, thus it depends on the definition of the relevant information that needs to be exchanged to the neighbors. Furthermore, the performance evaluation of algorithms to address the benefits in terms of functionalities and efficiency, and for the determination of the costs associated are essential for the process of development. Thus, we propose mechanisms for bootstrapping and discovery, to ensure the proper base following the Hide and Seek concept, where the nodes change their role dynamically according to events in the network, and with dynamic probing intervals according to the number of Seekers entering or leaving the network. The interactions in the Hide and Seek discovery are made at 1 hop distance in this initial stage, sharing information such as identifiers (MAC and Internal control identifier), network status, local resources available (% of free CPU and Memory), source and destination IP addresses for the direct connected nodes. However, 1-hop distance is not sufficient to provide a communication base for distributed management approaches. Thus, for an efficient exchange of management information beyond the limit of 1-hop nodes, it is important to consider the directions through which direct nodes or next hop relay nodes will be chosen to continue the discovery and exchange of management information as pursued in the Research Objective II.

1.2.2 Research Objective: II

How to ensure the exchange of partial information between non directly connected nodes?

The type and amount of information exchange that will be used to perform management and decisions beyond the communication process is very important. The distributed network management also defines the knowledge management required to be exchanged assuming optimized models and partial information about the network. Emphasis is given to the chosen relay nodes to continue the exchange of management information in the network, and to decide to which node(s) the information is further sent to. We argue that node cooperation is a key part of distributed management process, requiring periodic exchange of management information between the nodes. Moreover, a lightweight dissemination of management information is one of the major challenges in dense wireless scenarios. The distributed management approaches are characterized by completely eliminating the centralized coordination device, being crucial to assess how information is disseminated on a network built with devices, acting as both relays and end-points. To increase scalability of the network information exchange, network nodes need to cooperate through an efficient set of rules, policies and criteria, to minimize the time and amount of control messages flowing in the network. Thus, we are aiming to propose a lightweight solution to perform exchange of network information through the Neighbors Eyesight Direction (NED) function, enforcing cooperation between the nodes at larger management areas, and also including choosing the best neighbor node to forward the discovery process and exchange of information according to the knowledge depth of the network.

1.2.3 Research Objective: III

How to maintain, update and disseminate management and data information in highly dynamic environments?

Cooperation between the network nodes aim to periodically update the network management information, and do so independently of the network topology. This requirement aims at reducing disconnection between nodes, and optimizing the information exchange between them. The mechanisms required to operate when a node enters and leaves the network, the identification of the registered members and the association model, as well as the manner in which the nodes communicate with other members of the network, are important network issues to be addressed. These issues entail monitoring and fault detection functionalities, and maintain the consistence of management information between the devices whenever possible, especially on top of unstable scenarios such as wireless networks ad-hoc networks. The fault handling establishes communication feedback with the network administrator on the failures that affect devices, links and the network - possibly leaving the device incommunicable. Devices and links may fail for several reasons, such as devices hardware overflow (e.g. CPU and memory exhausted), the departure of nodes from the coverage area, bandwidth bottlenecks or highly congested links. Thus, a network must adapt itself to constant traffic and connection changes. We propose and develop a mechanism for association and connection between wireless nodes in communities through social-based metrics, in order to group the wireless nodes in communities and perform nodes' association. By the definition of new social metrics, the nodes are self-organized into quality communities which improve their message exchange for the support of management decentralization in wireless networks. These social metrics measure and predict the Friendship of Neighborhood Nodes, Associated Nodes and Community Nodes, so that the connection between the nodes is performed through this set of metrics to optimize their quality/stability and information dissemination efficiency. To improve data dissemination, optimization techniques using genetic algorithm are proposed using only technical information at MAC layer. To consider the social networks and interested users, we add a new factor that will infer the user social network preferences, enforcing a complete Device-to-Device (DtD) behavior which brings the nodes affinity at the network level, and thus will improve data dissemination in the network.

1.2.4 Research Objective: IV

How to provide proper management information to autonomic decision systems?

It is required to define the communication process to disseminate the local decisions to provide global cooperative decisions. The final decision is disseminated to the nodes that will need to enforce it. It is also important to define which nodes need to receive the information to provide the enforcement, which nodes need to access the information, and how to identify them to optimize the dissemination process. A high degree of correct decisions has to be obtained, even under the assumption that a node can only maintain some of the network information through partial management information and perform decisions based on this incomplete information. In addition, the support for multi-network technologies and environments is provided.

1.2.5 Research Objective: V

How to integrate the objectives I, II, III and IV into a communication infrastructure to support self and autonomic management over heterogeneous environments?

The practical way to integrate a set of optimized mechanisms in a comprehensive infrastructure for communication is through a communication framework. This framework contains all the necessary modules to ensure an optimized basis for communication between the nodes, working on multiple network technologies. The framework supports a distributed management plane inside the network, not in dedicated or external servers. Thus, it considers high collaboration between management processes, and the management processes embedded in each network device over a management domain. Each network management process interacts with the neighborhood through an overlay association. The communication is performed through peer-to-peer interaction, and relies on different propagation schemes to enforce management processes that allow a level of real-time awareness notification and reconfiguration in response to node addition or failure. The framework needs to be multi-platform, highly responsive to failures and anomalies in the network, and lightweight in terms of overhead impact and portable functionalities on top of multiple architectures and communication technologies (e.g. wired and wireless scenarios).

1.3 Contributions

The contributions of the Thesis are related to the articles developed during the research schedule. The papers compilation follows up the research achievements and describes the main original work contributions. With this purpose, the contents of most relevant publications inside each particular topic have been considered. Thus, this document is largely supported by the publications presented in the Table 1.1.

The main contribution of this Thesis is the definition of a communication framework for distributed network management aiming to create a lightweight communication infrastruc-

Topic	Paper	Title and Reference
Initial Communication	I	A Cooperative Hide and Seek Discovery over In Network Management [GVMS10]
	II	Nodes Discovery in the In-Network Management Communication Framework [GGP+11]
Exchange of Management Information	III	Lightweight Discovery and Exchange of Network Information in Distributed Network Management [GGS+13]
	IV	On the Analysis of Dissemination Management Information through an Eyesight Perspective [GCG+13]
Network (Re) Organization	V	Self-Organizing Decentralized Wireless Management through Social-Based Metrics [GGSS12b]
	VI	Improving MAC Layer Association through Social-Based Metrics in Mobile Networks [GGSS12a]
Data Dissemination	VII	A Spanning Tree Protocol over Mobile Wireless Ad Hoc Networks [GGSS13b]
	VIII	A Genetic Algorithm Approach to Improve Network Nodes Association [GGSS13a]
	IX	Multimedia Streaming Dissemination through Cooperative Device-to-Device Behaviors in Mobile Networks [GGSS]

Table 1.1: Publication Contributions related to the presented Thesis.

ture to support autonomic and self-management approaches. Such framework includes a set of communication mechanisms which enables the nodes to communicate and operate under a set of new concepts and stringent rules. The mechanisms are addressed at each stage of the communication framework, which are related to *Initial Communication*, *Exchange of Management Information*, *Network (Re) Organization* and *Data Dissemination*. New concepts are introduced and evaluated which enable the nodes to cooperate and collaborate, taking advantage of completely distributed scenarios as envisioned in the NGNs. The papers presented in the Table 1.1 also serve as basis for the preparation of Chapter 3 with the greater part of the information presented so far. Additionally, the Papers III, IV, VI and IX are the ones selected to be in the Annex A, B, C and D of this document.

Therefore, the interaction and the main elements that build the communication framework were first presented in the Paper II, showing the feasibility of the proposed communication framework for an operator static scenario in a distributed manner.

Regarding to the Papers I and II, they comprehend the initial communication stage of the proposed framework. They present the mechanisms for bootstrapping and discovery, ensuring the initial basis for communication between nodes, where the nodes share initial information between them. The original contribution of these Papers is the Hide and Seek discovery, where each Seeker will find each Hider according to a direction function. The most important characteristics detected and stressed in these Papers by analysis of the proposed mechanisms converge to: (i) network elements can change their role on the network (Hider or Seeker) dynamically according to the network situation and events; (ii) dynamic refresh intervals help to reduce the impact on overall network overhead; (iii) optimizing the internal event triggers according to the type of entity to be contacted is useful to ensure the proper network actions.

However, it is important to extend the discover and exchange of management information beyond the limit of the directly connected nodes, as pursued in the Papers III and IV. The main concepts for communication between nodes, as the case of the discovery by the concept of

Hide and Seek are maintained. Additionally, these Papers address the exchange of information larger than 1 hop performed by the NED function. To this end, a new functionality is proposed and evaluated as a complement to Hide and Seek method. The main contribution of this integration resulted in the protocol Hide and Seek Directional Discovery (HISK2D).

Regarding to the Papers V, VI they present the mechanisms for network (re) organization, which demonstrate the need to address mechanisms that work in the layer 2 (MAC) of the protocol IEEE 802.11. The innovation point of these papers are the presentation of the mechanisms for the association of nodes through social-based metrics. The social-based metrics are weighted by the technical factors aiming to infer the best association between nodes and their respective communities, which is demonstrated by combining the mechanisms of association and organization of the nodes through communities of association. We show that it is possible to measure and infer and organize the nodes through associations at level 2, reducing disconnection between the nodes and also exchanging of management information in communities generated in a totally decentralized way.

Papers VII, VIII and IX present the mechanisms used to provide data dissemination, more precisely, multimedia streaming content. The Paper VII addresses the development of a mobility-enabled baseline and loop-free extensions to the Spanning Tree Protocol (STP) for wireless ad-hoc environments. This baseline is used to complement the results of the Paper VIII and IX, proving the feasibility of the mechanisms front to well-known protocols established in the literature, such as the case of Wireless Spanning Tree Protocol (WSTP). Paper VIII presents a method based on Genetic Algorithm (GA) to determine the input combination of weigths that lead to the best solution for the technical social-metrics weights. Moreover, the WSTP baseline jointly with the optimization through GA aim to provide a complete base to assess the mechanisms for network (re) organization and data dissemination. Paper IX presents a novel perspective to disseminate multimedia streaming merging technical social-metrics with social network behaviors to better perform device affiliations, introducing the device-to-device (DtD) behavior. Those technical, originated from social-based metrics, plus a new factor which infers social affinity from the users' social networks accounts (e.g. Facebook, Google+, Myspace, Twitter, etc) enforce the nodes affinity at the network level, thus, enhancing the dissemination of multimedia streaming over mobile networks.

Synergies between research areas of this Thesis and the work proposed and developed so far, allowed the author to participate and contribute to different national and international research projects:

- The FP7 4WARD Project: In-network Management Design (Contribution: FP7-ICT-2007-1-216041-4WARD / D-4.3|2010-07-15) and Evaluation of the in-network management approach (FP7-ICT-2007-1-216041-4WARD / D4.5|2010-06-11). In this project, the Hide and Seek discovery was evaluated, aiming to discover network resources and capabilities for adaptation of management operation to current network working conditions.

- PANORAMA Project, funded by the Quadro de Referência Estratégico Nacional (QREN) through a strong collaboration with Portugal Telecom Inovação (Abril 2009 - February 2011). In this project, the communication framework through Hide and Seek discovery was assessed, where the communication framework was embedded in the network elements to provide 1-hop low overhead management information for the autonomic decision system in operator networks.

- SELF: Discovery and Autonomic Path Decision, working in the area of self-awareness, self-management and self-decisions over the top of operator networks (April 2011 –

March 2012). In this project, the communication framework including the Hide and Seek discovery integrated with the NED function was assessed in order to provide management information at larger than 1-hop, aiming to perform enhancements in respect to Hide and Seek discovery as well as the integration of autonomic decisions system for path reservations, and their extension to encompass distributed and autonomic management for network optimizations.

- UbiquiMesh Cross-Layer Optimization in Multiple Mesh Ubiquitous Networks, PTDC / EEA-TEL / 105472/2008, working in the area of Quality of Experience and Cooperation/Competition between different mesh operators (February 2010 – December 2012). In this project, the communication framework was assessed to establish a mesh operator-supported and distributed management platform through a multi-technologies demonstrator, unifying multiples technologies services and applications over wireless platforms.

Finally, the technical material developed in the scope of this research work, followed open-source rules for development. The author collaborated with one M.Sc. student and two technical researchers, which resulted in a prototype written in C/C++ (HISK2D). The author also helped to analyze and assess the simulations using Network Simulator v.3 (NS-3) developed by the M.Sc. student. Additionally, the author performed numerical simulation through MATLAB, which helped him to analyzes and understand the behavior of the proposed mechanisms on top of wireless and scalable scenarios.

1.4 Thesis Outline

The topics included in the scope of this Thesis entail self and autonomic management approaches, mechanisms for communication between the nodes, including the bootstrapping, discovery, exchange of management information, nodes association and organization and data dissemination.

The Thesis outline, containing all the Chapters involved, where each area possesses its own set of mechanisms, algorithms, protocols and problem statements on the subjects addressed in this Thesis.

As such, Chapter 1 presents the motivation for the research conducted in this Thesis, as well as associated challenges and contributions.

Chapter 2 provides a brief overview on network management principles, comprising the description of their key characteristics that compose classic network management system and application model. Then, an insight on the added value brought by network management architectures illustrates the approaches used, such as traditional, centralized, distributed and cooperative management. In the following, Chapter 2 dives into the topics of self and autonomic management for Next Generation Networks (NGNs), presenting a literature survey of several projects, solutions and architectures used to decentralize or distribute the control and management functionalities in the network, especially the ones that rely on self and autonomic concepts. Regarding the mechanism for node communication, an overview and explanation of the IEEE 802.11 communication and associations standard are given, helping to understand how the wireless nodes can associate and communicate efficiently. Thereafter, an overview of approaches for network bootstrapping, nodes and topology discovery, exchange of management information, nodes association and network organization is presented, showcasing their limitations, drawbacks and disadvantages. To conclude the state of the art survey, Chapter 2 points towards the possible directions for future innovation in the field, and the ways it could be accomplished through the use of innovative concepts.

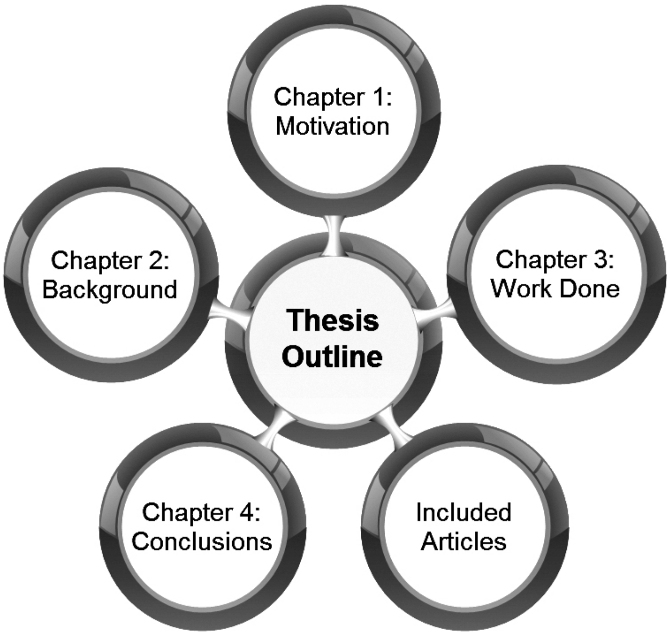

Figure 1.2: Thesis Outline and Chapters Sequence

Chapter 3 describes the proposed communication framework that integrates all the mechanisms developed in this Thesis. We start by contextualizing our communication framework from the device point of view, identifying the following main research topics: Initial Communication, Exchange of Management Information, Network (Re) Organization and Data Dissemination. Then, we explore each research topic through a more elaborated description of the concepts, functionalities, interactions and achieved results. The Initial Communication research topic addresses the proposal of a low overhead communication through bootstrapping and network topology discovery mechanisms. Thereafter, we specify the interaction between nodes in order to optimize the initial contact between them. In the scope of the Exchange of Management Information topic, we define the type of management information to be exchanged beyond the limit of the directly-connected nodes. It considers that a node can only maintain some of the network information and perform distributed management based on incomplete network information. (Re) Organization and Data Dissemination describes the way that a node, or a group of them, may be organized under dynamic and unstable situations as well as the enhancements performed in the data dissemination mechanisms in order to disseminate multimedia streaming content. Moreover, we include in this Chapter a brief description of proof-of-concept testbed in order to prove the practicability of our mechanisms for discovery and exchange of management information in real and heterogeneous networking environments.

Those topics resume our developed work, but each topic defines and addresses their own set of problems and challenges to provide a lightweight communication between nodes for autonomic and distributed management. The mechanisms proposed were devised to fit the most diverse conditions and events, such as device and link errors, different traffic loads and network requirements, different standards (IEEE 802.3 and IEEE 802.11) and network technologies (wired virtual grids, wireless mobile ad-hoc networks and large-scale wireless networks). Finally, we conclude the Chapter highlighting the main innovation points and applicability of our proposed mechanisms in order to meet the requirements posed by the NGNs.

To conclude, Chapter 4 summarizes the overall outcomes of this Thesis in the scope of communication between nodes for autonomic and distributed management, through a set of results and achievements according to the research objectives. Moreover, the envisioned future work points towards adapting the proposed mechanisms to other research areas and scenarios, such as the case of Software Defined Networks (SDNs). Finally, the Papers III, IV, VI and IX from Table 1.1, compose the Papers in the Annex A, B, C and D, indicating some of the potential communication framework and the mechanisms involved as well.

Chapter 2

Background

This Section presents the state of the art on the key targeted areas addressed by this Thesis, composing i) Network Management Principles, ii) Self and Distributed Management Approaches for Next Generation Networks, iii) Mechanisms for Communication between Nodes. Finally, Section 2.5 describes problems and challenges of current solutions which have been contributing to identify the need to propose new mechanisms and evolutions in order to perform an infrastructure for communication between nodes.

2.1 Network Management Principles

Network management enables execution of functionalities for monitoring, controlling, planning, allocating, deploying, and coordinating computer and telecommunication networks. Normally, all those functionalities together compound the management functional tasks [SGR10]. A more formal network management definition is described in [Cle07]:

- Network management refers to the activities, methods, procedures, and tools that pertain to the operation, administration, maintenance, and provisioning of networked systems.

- Operation: includes monitoring the network to react to the problems and, when is possible, before the managed equipment is affected.

- Administration: involves observation of resources in the network and how it is configured as well.

- Maintenance: is concerned with performing repairs and upgrades. One good example is when a new switch or a router is added to the network and needs a new operating system image. In this case, the maintenance also includes preventive and corrective proactive, to allow adjustments that will make the managed network run in the correct way.

- Provisioning: deals with configuring resources in the network to support the services. It also includes setting up the network allowing a new customer to receive multimedia services.

A Network Management System (NMS) basically consists in managers, agents, network elements, Network Management Stations (NMSs), management protocols, Management Information's (SMIs) and in Management Information Bases (MIBs) as systematic roles, with both manager and agent functionalities [KV97, Cle07]. The manager is responsible for the coordination of management functions, while an agent performs simple tasks like collecting and recording management information of a network element. In this case, the number of sent/received packets, plus the amount of error packets received are examples of collected management information. Looking at this scenario, network elements are hardware devices such as routers, switches, terminal servers or computers, so any of these candidates are able to become a managed device. NMSs are consoles which execute management applications to monitor and control a network element. Generally, NMSs use Common Line Interfaces (CLI) to manage physical or logical resources of elements to be managed, and usually at least one NMS is an essential requirement to the classic management environments. The management protocol is responsible for the communication between agents and NMSs, SMI is the language used to define the rules as instances of the data, and MIBs are the hierarchical collection of management information objects (i.e. system, interface, address translation, IP, TCP, UDP, etc.). Those management collections can be used to analyze the behavior of a network element, which can help the network administrator taking necessary decisions in case of system malfunctions. The Figure 2.1 helps us to illustrate the classic NMS working and the applications that are managed in this process.

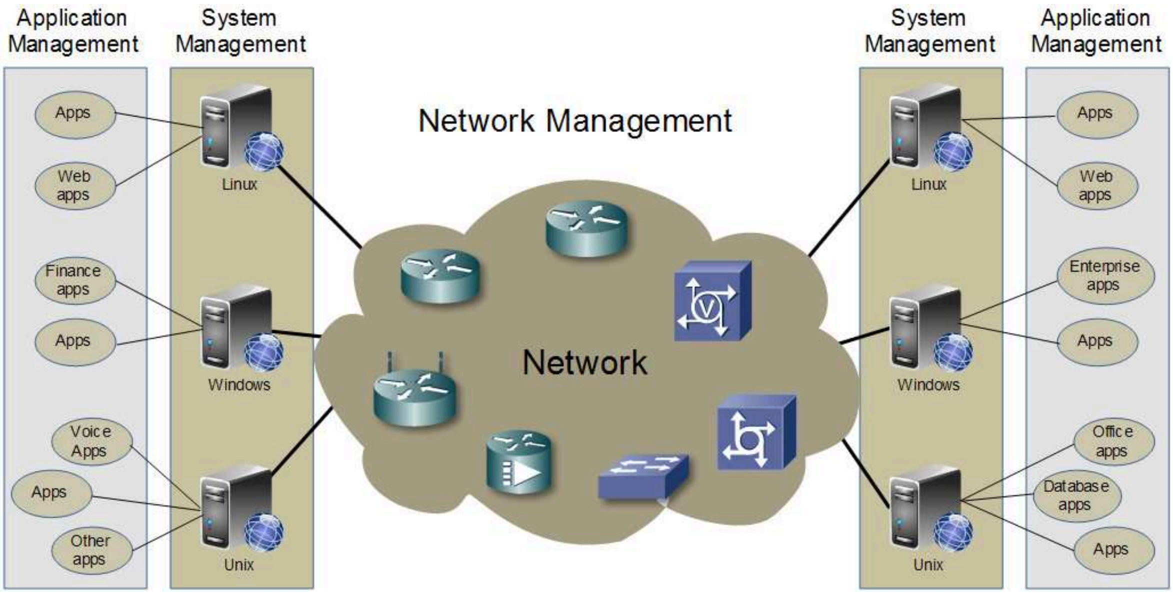

Figure 2.1: Classic network management system and application model, adapted from [Cle07]

The management application integrates the group of applications (e.g., office, web, database, voice, finance, etc.) that will be managed by the system management group, and all the system management components work over the network management as well. The system management uses dedicated servers with different operational systems (e.g. Linux, Windows, UNIX, etc.) and management functionalities residing outside of the network. The system management interacts with network management via management protocols such as SNMP or CMIP jointly with Fault, Configuration, Accounting, Performance, Security management (FCAPS) [ITU96, ITU97]. FCAPS is defined by International Standard Organization (ISO) Telecommunication Management Network model and defines the main telecommunication network management tasks. Fault management refers to the set of functions that recognize, isolate, correct and log faults that occur in the managed network. Configuration management gathers and stores devices configurations from network devices, in order to simplify the configuration of the device. The accounting management refers to the billing management process and provides the usage statistics of the users (i.e. disk usage, link utilization and CPU time). Performance management refers to monitoring network utilization, in terms of end-to-end response time, throughput, percentage utilization and errors rate). The security management supports functionalities that allow only authorized access to use the network (i.e. authentication, encryption and authorization).

FCAPS is useful to understand the requirements and goals of network management as well. On the other hand, regarding the management of telecommunication networks, we can have the accordance between the parties where the customers can become the service providers. A Service-Level Agreement (SLA) consists on negotiable agreements that offer different levels of availability, serviceability, performance, operation, or other attributes to the service, such as billing in the services provided [AIH97]. This level can also be specified as 'minimum' average of performance for the user. In addition, penalties can be applied in case of non-compliance of terms and it also depends on SLA contract type. This type of agreement is more common in Telecom services providers (i.e. Vodafone, TMN, Optimus, etc.).

2.2 Network Management Architectures

2.2.1 Traditional approaches

In [MFZH99], it is proposed a taxonomy for classification of traditional management approaches that are grouped into four main types: centralized, weakly distributed hierarchical, strongly distributed hierarchical and cooperative paradigms. This taxonomy will be used to better explain the traditional network management approaches, like it is shown below.

2.2.2 Centralized Approaches

In this approach the communication network is managed by a centralized system, which typically contains one manager. This manager controls a potentially large number of elements through local management agents, and the main idea consists in one manager who coordinates multiple agents (Figure 2.2). Most of computer networks are managed by SNMP that refers to a set of standards to networks management, including a protocol, MIB's structure and a set of data objects. The first release of SNMPv1 is the most popular and was adopted as a standard for TCP/IP-based networks, where almost every IP device had an instance of SNMP-embedded agent. A supplement to SNMP, known as Remote Network MONitoring (RMON), was proposed to extend the capabilities of SNMP to include management of Local Area Networks (LANs). Following this idea, the SNMPv2 RFC 1448 [MRW93] was created to enhance specific problems in the protocol operation of the previous version. The version SNMPv3 RFC3410 [MPSJ02] was proposed, to improve authentication, remote configuration and security features. Also, in the literature few works try to extend the SNMP for distributed networks [Bea00, Pea02], but it is still facing the problem of deployment cost, and the manager continues to work in a centralized way.

Regarding centralized approaches, the simple usage has a considerable number of disadvantages [MFZH99], which are:

- Manager introduces a single point of failure. This can be solved by creating redundancy, but it is usually a costly operation.
- The agent simply collects the management data and the logic is concentrated in the manager to analyze all the data obtained. In this case, the absence of collaboration between agents can be detected.

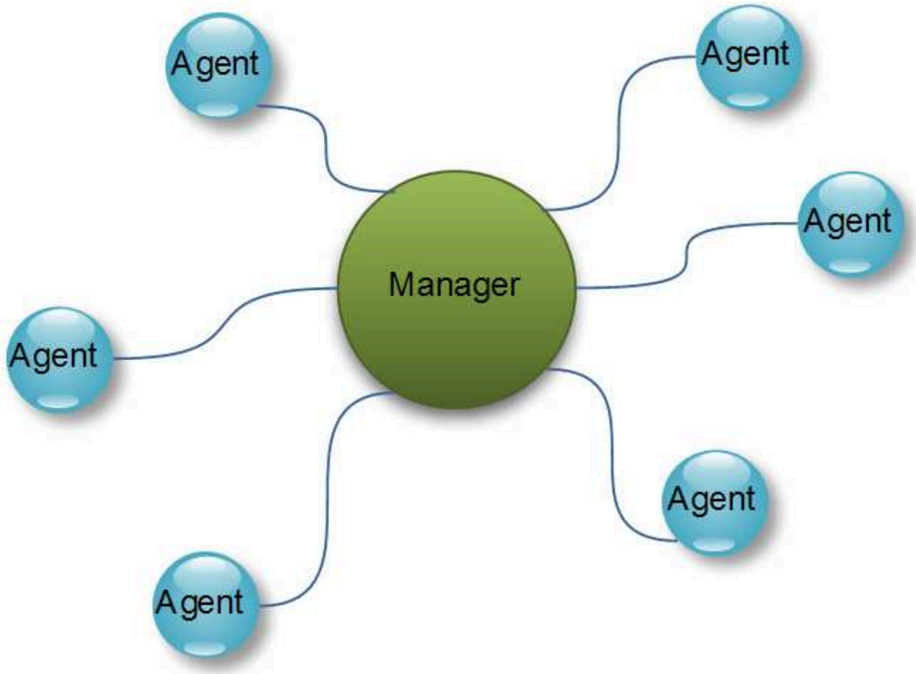

Figure 2.2: Centralized approach example.

- The use of a central entity (manager) has a negative impact on the scalability, due to the dependability of the NMS for every management action.

2.2.3 Weakly Distributed Approaches

This approach consists in distributed models where the management processing load is divided among the Agent Managers in the same hierarchical level. Here the communication between the Agent Managers is minimal and all communication needs to transverse the Manager, as depicted in Figure 2.3. The approach that uses weakly distributed model is OSI Management (OSI-SM) [Pav07] which uses the CMIP for communication between Manager and Agents. This CMIP is a management protocol that provides an implementation for the services provided by CMIS, and allows the communication between the network management applications and the management agents. On the other hand, OSI System management is also used as a base for the Telecommunication Management Network (TMN). TMN is defined by the International Telecommunication Union (ITU-T) as a framework for telecommunication management networks in ITU-T M.30000 recommendation series, and is widely adopted by telecommunication management networks. On the other hand, distributed object computing can be considered a weakly distributed approach. This process consists in software modules designed to allow multiple equipments, connected via a network or different types of processes inside, to work together at the same equipment. Until now, in this approach, most of the research has been performed using the Common Object Request Broker Architecture (CORBA) [LCT99, LPR97, Pav99] as a base, and the main idea consists on creating collaborative processes to allow the communication across the network or through the inter-process communication.

Regarding to weakly distributed approaches, there are a few number of disadvantages [Cea02], which are:

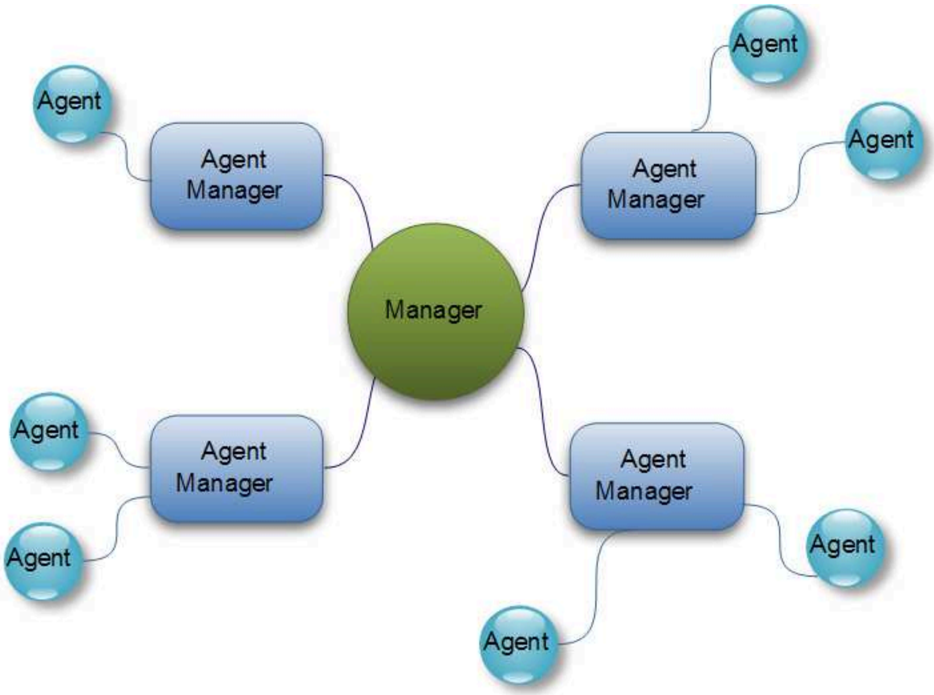

Figure 2.3: Weakly distributed approach example

- Similarly to centralized approaches, the agent simply collects the management data and the logic of the data is concentrated in the Agent Manager. After that, the Manager analyzes all the data without absence of collaboration between Agent Managers and agents.
- The hierarchical and static configuration of Agent Managers still implies a single point of failure and the redundancy could be equally expensive as described in centralized approaches.
- The lack of robustness in this approach, which can be observed when a connection between the Agent Manager and Manager is lost and the agent cannot take actions. Due to this drawback, weak approaches are not suitable for large and dynamic environments [MFZH99].
- Distributed object computing approaches require a dedicated and a real time support, and this may not be available in every network equipments.

On the other hand, in strongly hierarchical distributed approaches (as opposite to the weakly distributed approaches) there is a relationship between the Agent Managers, as depicted in the Figure 2.4. As previously mentioned in both approaches, agents have been working just as simple data collectors, and the devices can do much more than just running a simple agent, and could even be capable of managing themselves if a goal was given. The distributed task by delegation is an example of strongly distributed approach. The Management by Delegation (MbD) framework was proposed to demonstrate the potential structure of relationship between Agent Managers for task delegation, and the main characteristics that differ between weakly and strongly distributed approaches is in the fact that each Agent Manager distributes the decision along the other Agent Managers in the network. The Manager has the main function of coordinating all delegation tasks, but in most of the cases the distribution of the communication is done by the Agent Managers.

Moreover, other examples can be seen in [KB97] where most decentralized approaches for network management (Distributed Big Brother, Distributed Management Environment, Hierarchical Network Management and Management by Delegation) were compared in terms of polling method, communication between layers of management, extensibility and flexibility. As a result of this comparison, the distributed management approaches have their own specific advantages and disadvantages, depending on environment, but they still have problems in terms of communication costs, autonomy, extensibility, flexibility and robustness.

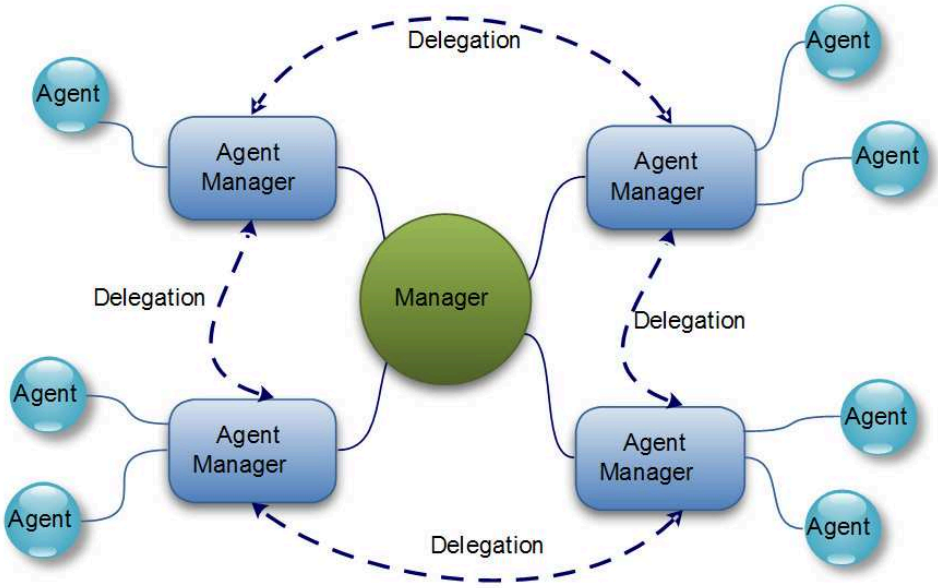

Figure 2.4: Strongly distributed management approaches example

Strongly distributed approaches overcome many existing problems in the previous two approaches, but it also contains some drawbacks [MFZH99, Cea02]:

- Mobile agents need the specific platforms or containers to migrate/delegate tasks along the network.
- Extra overhead for task migration is an imminent disadvantage of this approach.
- Mobile agents use proprietary and non-flexible platforms or containers which result in non-flexible implementation.
- The Agent Managers are subordinated to a Manager, and they do not take intelligent decisions by themselves.
- As in weakly approaches, the single point of failure is maintained.

2.2.4 Cooperative Approaches

In these approaches [ZLBY07, KLS97, BPW98], like the title suggests, the cooperation is the main characteristic of the management system. A cooperative model works with a large amount of roles among the entities. These entities act in both managers and agents, and work together in a collaborative way fulfilling management tasks. An important property of this approach is the fact that collaboration and communication are not restricted to a fixed managed point. The agents can enhance the autonomy and intelligence, and mobility enables collaborative and distributed tasks across the network. There are two main categories in agent-based management techniques: the mobile [BPW98] and the intelligent agents [KLS97, BPW98]. By the agent-based management we can decentralize the processing and control, thus improving management efficiency, as depicted in the Figure 2.5.

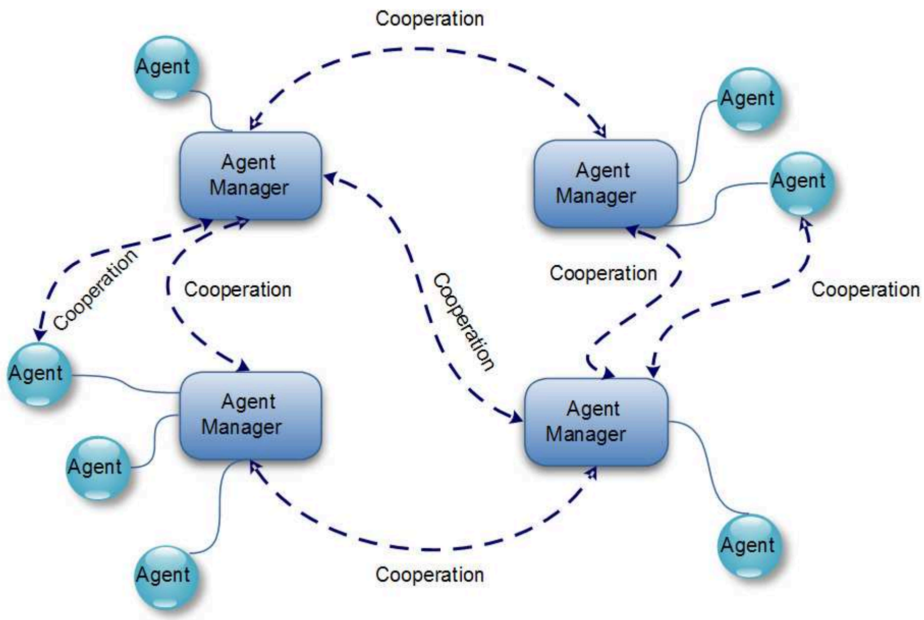

Figure 2.5: Cooperative management approach example

An other example of cooperative management comprises the Active Networks (ANs). AN's management enables network elements, such as routers and switches, to be programmable and perform higher flexibility to network functions, like for example the routing process. Active networks combine code mobility enabling technology for distributing management tasks to the device level [HMZ08, MHZ08]. Active network management uses proactive management, which operates on the ground of symptoms predicting negative events in order to avoid them.

However, a few number of disadvantages in cooperative agent approaches are presented:

- High degree of intelligence makes this approach more complex to implement.
- Absence of security and performance in large-scale scenarios.
- High cost per device to collect, store and act in the management environments.
- The Agent Managers are delegated manually by network administrator.
- Each agent is responsible just for collecting management data, and the intelligence of decision is made by Agent Manager.

- Active Networks use proactive management concept. It turned out to be inefficient in constant changing environments.

In the following Section 2.3, autonomic and self-management approaches are presented in order to overcome the limitation imposed by traditional management approaches.

2.3 Self and Autonomic Management Approaches for NGNs

The main challenge of NGNs is the efficient management of increasing complexity of services and applications. Future networks must be flexible and scalable, easily configured and updated. The idea of Autonomic Computing (AC) is derived from the body's autonomic nerve system that controls the main functions without an entire involvement of all other parts. AC was an initiative started by IBM in 2001 [AutaN] and consists of a systematic computer-based approach that enable systems managing themselves without human intervention in the whole process for middle and low level decisions. The goal of autonomic network is to increase the management task automation by the means of autonomic processes.

An AC system has four basic properties for self-management. These properties are known by Configuration, Healing, Optimization and Protection (CHOP) [BJM+05, KC03, MOKW06, Dar09]. Self-Configuration frees the administrator to adjust properties according to changes in the environment, Self-healing automatic discovers and corrects faults; self-optimization automatic monitors and controls resources utilization according to the defined parameters [CMAB13]; and self-protection frees the administrator to secure the system against malfunctions provided by security attacks. The main idea of self-management is to manage complex environments, adapting to unpredictable conditions, prevent and recover from failures and provide safe environments [Gup06] by the means of Self-X properties. Also, at least, an AC should have three main features [Din10, DDF+06]. First, an AC should be more self-aware: it means that each element in the network should have a detailed knowledge of its surrounding elements in order to act accordingly to its knowledge. Second, an AC should have the capacity of self-configuring and self-optimizing, and must also configure and reconfigure according to varying conditions. Third, an AC should be capable of self-healing, which enables it to recover from malfunctions parts. In [Din10] the author suggests that Auto-agnostics is a new functionality paradigm that describes the capacity for any entity or computer network to be more self-aware. Auto-agnostic includes a range of self-discovery, awareness, and analysis functionalities that help an AC to be more self-knowledge, as well as its surrounding entities in order to gather, analyze, and report management information.

Recent initiatives in European Telecommunications Standards Institute (ETSI) were proposed that aimed at standardizing resilience, survivability, autonomic fault-management and autonomic security management [CWBM+13]. The vision of self-management is seen as an alternative to the traditional management of a network with large size, variety and dynamic of elements [JvB+07, FKALG+06, LST13]. In ETSI, AC consists of a control loop [AutaN, KC03, Mil05] that controls the managed elements. The control loop is a set of components that are responsible for monitoring managed entities, acting and collecting relevant data information and assisting in high decision making. This data information is continuously collected and analyzed, and afterwards compared against a desirable state. An autonomous element is an agent who manages the internal behavior and its relations with other agents, respecting rules and policies. Therefore, these control loops will influence each other as a whole. Some direct and indirect changes in a subsystem influence numerous separated control loops, where it is difficult to maintain the consistency and accuracy of decisions as well. An example of autonomic control loop is depicted in the Figure 2.6. Monitor or collect, analyze or inspect, decide or plan, and act or execute, is the correct order of the autonomic control loop.

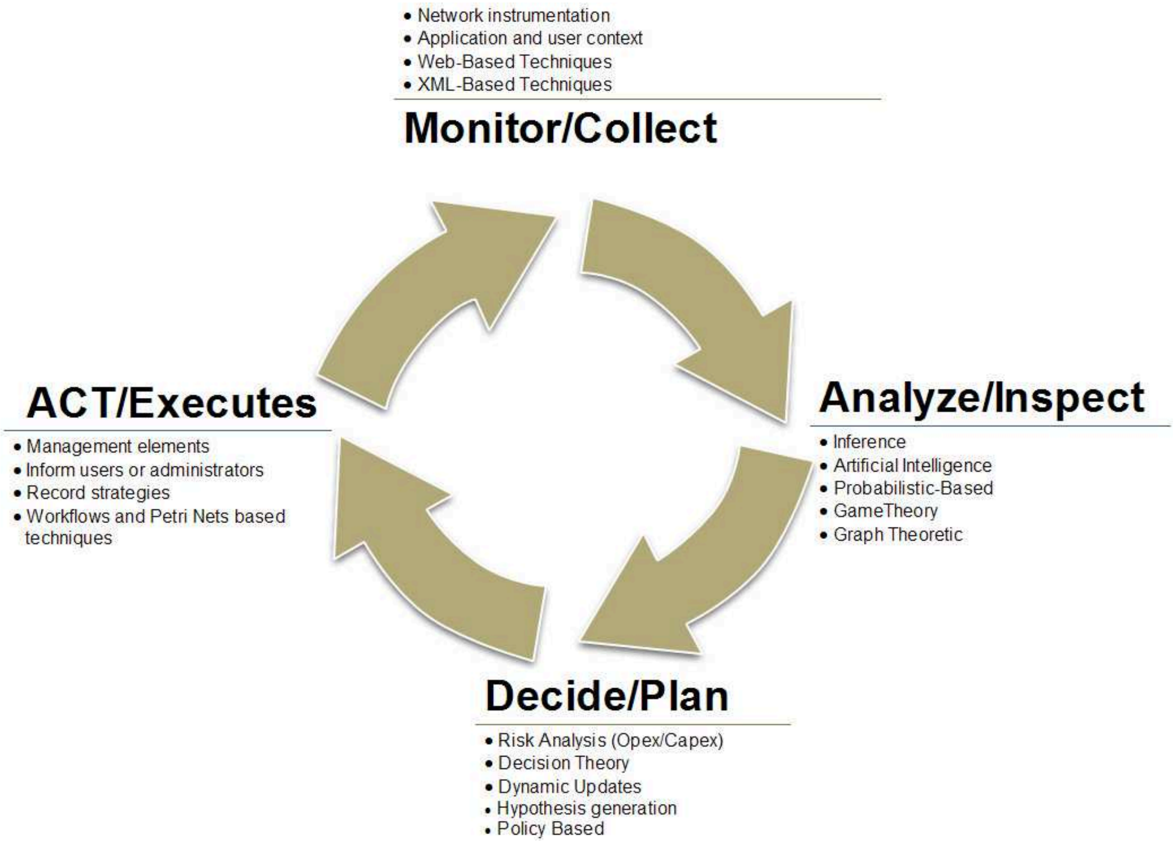

Figure 2.6: Autonomic Control Loop, adapted from [AutaN, KC03, DDF+06]

Monitor or collect provides theories, mechanisms and techniques to collect, merge, aggregate, filter, and report details collected from managed resources. The monitoring and collecting of management information can be web-based network management. Web-based network management has the ability to monitor as well as manage a network, not considering the location of the network or the network manager. It uses the Internet as an access base and basically consists of three network management system frameworks: centralized, hierarchical, and distributed [AIH97, HG98, YWW99]. However, web-based network management suffers the problem of integrating the Graph User Interface (GUI) with lower level management commands. Also, there are some proprietary solutions to monitor managed networks [Sol11, IBM11]. However, these approaches are not flexible, and it is necessary to pay royalties for their utilization. In this case, Open Source approaches are desired [Gro10, KPP+13].

Analyze or inspect functions provides the mechanisms, theories and techniques that correlate and model complex situations. These mechanisms allow the autonomic network to learn about the problems and malfunctions, as well as predict future situations. Machine learning, artificial neural network and expert systems are some examples, which can be extended to rule, case and model based systems. The main advantage is the variety of approaches that can be used to analyze complex problems over network management. In addition, bio-inspired management approaches are an other example close to artificial intelligent management approaches. The main difference is that the intelligent management uses more evolutionary concepts to perform learning and reasoning. On the contrary, a bio-inspired strategy uses abstraction of behaviors found in the nature [PM09, Wil08], e.g. behavior of ant's colony and immune systems. However, bio-inspired approaches face the same problem related to artificial technique, where the designers need to understand all interaction processes of the model to be abstracted as well as avoiding external complexity on the management.

In probabilistic management models there are various situations where uncertainty confuses management system developers. The probabilistic models are well used when the management system designers have no exact knowledge of the managed network in which their system is supposed to work.

On the other hand, an efficient network management system should deal with uncertainty proposing alternatives to management tasks [BDMN09, BSF06]. Basically, the probabilistic approaches using Bayesian networks and Fuzzy Logic [SK08] are the most popular models. Bayesian Networks (BSNs) are probabilistic knowledge models that represent the cause-and-effect relationship between the entities in the management system. Bayesian networks can be used to predict future faults and support decisions even in the presence of uncertain or incomplete information. Bayesian networks use Direct Acyclic Graphs (DAG) [Mat10], with probability labels that represent all probabilistic knowledge.

Some applications for fault management have been proposed [BPM+10, DLW93, DKBC05, WS93b], and most of them provide an approach to fault diagnosis and fault location. In terms of Fuzzy logic approaches, these are composed by multi-valued logic derived from fuzzy set theory. On the literature there have been several proposed approaches [GRSC98, ICH08], and one of these uses fuzzy-logic models for temporal reasoning phase of the alarm correction process in fault managed networks [AD99]. However, probabilistic management approaches are a topic not fully explored by the research community, and it requires more investigation to increase the accuracy of these management models.

Decide or plan provides functions, mechanisms, algorithms and techniques that build the actions needed to improve network operation. Normally, the planning techniques use policy-based information to guide its work. The Policy-Based Management (PBM) [CS03] is a combination of rules and services, where the rules are defined according to criterions of resource access and usage in the internet. Moreover, the policy based-management is a promising solution that can be applied on large scale-scenarios, also in adaptive systems that dynamically change their behavior constantly. However, for a correct system operation, all the policies and rules must be well defined by the administrator of the network which not always occurs. In addition, the policies dissemination mechanisms [Zhi09, SVBM08, NNPB10] should avoid excessive message overhead and support collaboration and cooperation between managed entities in the network.

Other management techniques to decide and plan can be presented in terms of economy theory [Fea06], finite-state machines [WS93a], model traversing [JP93]. The Extensible Mark-up Language (XML) [W3C03] based techniques, provide mechanisms for organizing and transmitting complex information that will be used to manage the network. The main advantage is to apply the standard offered by XML over the communication between managers and agents in a network management environment, and can be used to organized the collected information. However, XML is a set of tools that helps designers to standardize management communication; due to variety of XML standards, sometimes it is not so easy to reuse the content in other communication applications. In addition, some investigation is related to applying game theory techniques for decision in radio resource management mechanisms on a heterogeneous environment [CMT08]. Therefore, these techniques raise the necessity to develop research efforts towards the maturation level of models and theories by the researchers' community.

Act or Execute provides functions, mechanisms and techniques that control the execution of a plan, taking into account dynamic updates. Petri Nets [Pet77] and Workflows [GKW+12] are examples of the most prominent techniques that can be used in order to comprehend the execution of management strategies or plan. In [MdX09] the authors propose a solution named PACMAN system, a Platform for Automated and Controlled network operations and configuration MANagement. PACMAN realizes network execution adopting Petri nets to manage the action and execution of management activities. However, the main aim of a Petri net is to provide concurrency for the execution of specific tasks, the high level of synchronization between the actions that will be executed; the tasks where they are executed are the main problem of these approaches. In order to provide accompanying of executing task, the authors in [MB08] propose to design and develop mechanisms for building an autonomic workflow management system that will incorporate the properties of autonomic computing and exhibit the ability to reconfigure itself to the changes as well as discover, diagnose and react to the disruptions through workflow execution, and monitor resources automatically. However, this work is limited to the grid environments scope.

To sum up, the main essence of autonomic computing was presented. The above mentioned components of the autonomic control loop can communicate and cooperate with other components in order to achieve autonomic behavior in each managed element. Afterwards, a trade-off between many theories, mechanisms and techniques that compound the autonomic control loop need to be well-defined, in order to ensure the desired autonomic behavior in the management systems.

2.3.1 Autonomic and Self-Management

The current tendency in terms of self-management is related to management of Future Internet. In fact, the Internet has doubled in size in the last decade. This tremendous growth has brought on an accumulating complexity in management tasks [BDMN09, GKP+09, RPK+09, NBEK+08, DBN+09, KCJ+08]. In this sense, it is extremely useful to use the clean slate approaches to solve open issues. In the following we present some related projects that use different technologies, strategies and techniques to ensure autonomic and self-management in different layers, scenarios and applications. Note that most of them are European Initiatives.

Autonomic Service Architecture (ASA) is introduced in [FKALG+06], and consists of a framework for management of Internet service and their resources. ASA framework addresses both service and network management introducing different levels of abstraction. The authors consider that everything is a service and all services are organized into a service hierarchy. Using this perspective, ASA ensures an uniform framework for service and transport network management. The design of ASA enables autonomic management of resources according to SLA between service providers and customers. Regarding the entity, in ASA each management entity is called Autonomic Resource Brokers (ARB). Each ARB is the analogy of IBM's autonomics element [AutaN] and is responsible for managing services, which are organized in a hierarchical structure and interact with virtual resources and other ARBs. In this research, the authors show an illustrative example on how ASA can be applied to the management of DiffServ/MPLS based on bandwidth sharing scheme. This mechanism can be automatically adjusted according to measured traffic load and efficient resource utilization accordance with SLA compliance. The ASA is characterized as a generic architecture, which encompasses different abstraction layers and heterogeneous resources. However, ASA has a high level of complexity in the hierarchical structure of the management entities, and also in the internal entities scheme. The main problem of this architecture is when new functionalities are created, and it can be difficult to engage them at each entity.

Autonomic Network Architecture (ANA) is an initiative established by European Foundation with the aim to develop an architecture for autonomous networks [ANA06]. The architecture allows the dynamic adaptation and reorganization of traffic according to the user's requirements. Moreover, the architecture supports mobile nodes and multiple administrative domains. In the architecture of each individual node, the application layer provides services such as WWW, E-mail, FTP, DNS, to peer-to-peer (kazzaa, emule), voip (Skype), among other supported applications. The concept of compartment is also introduced in the architecture of a node, and it is implemented according to operational rules and administrative policies for a given communication context. The approach defines the mechanisms required to operate when a node enters and leaves the network, which are the registered members, the authentication model, and how the nodes communicate with other members of the network (routing, peer-to-peer communication, addressing). ANA does not need to manage an unique global addressing scheme and to impose a unique way to resolve names, which becomes an advantage as well. However, ANA faces the same problem related to ASA architecture in terms of high level of complexity in hierarchical structure of the management entities and in the internal structure.

The CASCADAS Component for Autonomic, Situation-aware Communications and Dynamically Adaptable Service project uses the model of Autonomic Communication Element - ACE. An ACE acts as an entity that can implement a distributed way of communication and services, and also as a central point of service [MZ06]. An ACE model must integrate all self-organized capabilities and must be the basis for the implementation of the situated services of communication in the application layer, as well as the basis of implementation for the network and middleware layers. In CASCADAS, the application layer distinguishes the dynamic development of components from the coupling of components for autonomous applications. It contains capabilities such as self-healing, pervasive supervision and knowledge of the network, that are dynamically aggregated functions according to the autonomous behavior of the network. The network layer contains the integration of different devices such as personal television sets, personal computers and even sensors. Moreover, the issue of sharing computational resources is also addressed. The main advantage of CASCADAS is to define the underlying technology for a new highly distributed pervasive service that addresses the complexity problem at the level of resources and services. However, due the fact that an ACE works as a central point of service, this characteristic implies scalability problems in a large-scale network.

On the other hand, Haggle [HAG06] is a project initiated by European Foundation that consists in projecting an architecture for autonomous networks that support the presence of intermittent connectivity, that means, it explores the opportunistic communication. The node architecture provides all the aspects of communication, from the physical layer to the application layer. In the physical layer, we can find some methods of radio frequency supported as 802.11, Bluetooth and wired networks such as the Ethernet (IEEE 802.3). The architecture of a Haggle node is based on the notion of Haggle Information Space (HIS) having Application Data Units (ADUs) on each users device. The classifier has the role to classify the information that arrives from one of the inferior layers. Therefore, Haggle does not support end-to-end transmission, which turns out to be a disadvantage when it is required to provide quality of service and applications.

On the other hand, Foundation Observation Comparison Action Learn Reason (FOCALE) [SAL06] is a distributed architecture where each network element can incorporate the autonomic management functionalities. The main goal of FOCALE is to provide an architecture that adapts to changes in the environment, business rules and user requirements. FOCALE is a distributed architecture, so that it foresees that each network element can incorporate the autonomic management software. The FOCALE system is characterized by a high level of autonomy, in which the human intervention is only required in a business goal level. In [JvB+07], a novel autonomic management methodology, based on business concerns and technical problems is presented. Review about the benefits of autonomic systems is also used as a base to justify the importance of autonomic management system that combines functional architecture FOCALE with an information model (DEN-ng) to realize a distributed software system. This system controls and maintains a heterogeneous and highly dynamic environment, which is itself constructed using different software packages/products running on different nodes, using different operating systems, communicating using a variety of protocols. However, this system is very complex and difficult to recover in case of failure of the management system. In case of unpredictable changes in the environment or development of new functionalities blocks, it also becomes difficult to make modifications in the system and guarantee overall entities to work as expected. The complexities can also impose a negative effect in terms of management security and quality of service.

MANNA Management Architecture for Wireless Sensor Networks targets specially to wireless sensor networks. The MANNA architecture utilizes procedures that abstract the vision of the system [RSO+04]. These procedures execute management functions according to management policies. The dissemination of management functionalities that include policies and rules distinguishes between manager, agent and management information base or MIB. In this project, the manager can be centralized, decentralized and hierarchical, according to the application running on the wireless sensor. The project design takes into account the specific characteristics of wireless sensors networks such as storage, energy and battery consumption, faults and self-organization. However, Manna architecture is considered to be limited just for wireless sensors networks.

Autonomic Internet Autonomic Internet (AUTOI) [Aut08] is an architectural model consisting of distributed management systems, which are described with the support of five abstractions, namely the OSKMV planes. The Orchestration Plane, Service Enablers Plane, Knowledge Plane, Management Plane and Virtualization Plane (OSKMV) planes are distributed functions that control the network infrastructure. The orchestration plane is a conceptual definition for the functions that govern and integrate the behaviors of the management systems distributed across the network. This plane supervises the optimization and the distribution of knowledge within the Knowledge Plane to ensure that the required knowledge is available in the proper place at the proper time. The Orchestration Plane would host one or more Autonomic Management Systems (AMSs). The knowledge plane consists of models and ontologies to provide improved analysis and inference capabilities; its intention is to provide knowledge to enable the network to be self-monitoring, self-analyzing, self-diagnosing, and self-maintaining. The Management Plane consists of AMSs, and the majority of management functionality should be embedded in the network. However, the high level of synchronization turns out to be a disadvantage on the use of AUTO-I in large-scale networks. In [SZN+07] the authors apply autonomic and distributed management principles to wireless access networks, including adaptive, aware and automatic operation in a decentralized setup (such as GERAN/UTRAN, E-UTRAN, WIMAX or WLAN) by developing the autonomic management module that interfaces the legacy API. They also believe that the architecture can be used in other network management scenarios, like cellular networks, once the work evolved to UTRAN network. However, some limitations in other autonomic systems using this technique, related to the accuracy of the wireless measurements or the scanning module, can be also detected. The authors in this work highlighted that the instability of the results could be caused either by a limitation of the wireless-tools library by a WLAN interface driver or by the hardware itself.

In [FDT13], a federated architecture to autonomously manage multimedia delivery networks in a scalable manner was proposed. The main contribution consists of a framework to negotiate and configure federations between the stakeholders involved in the delivery of multimedia services. The results showed that including additional actors, such as cloud storage providers, in a multimedia delivery federation can greatly increase efficiency and decrease costs.

Initiatives to management through autonomic and scalable management for private clouds can also be seen in [SFO+13]. In this work the authors proposed Snooze, which is an open-source scalable, autonomic, and energy-efficient virtual machine (VM) management framework for private clouds. The authors presented experimental validations of two key properties required by Cloud resource managers: scalability and fault-tolerance.

In [YB06] the analysis of using autonomic computing to manage next generation services can help to optimize the communication and to understand about the controlling in autonomic manner, being useful to avoid the complexity of existing service management systems. In this case two views must be analyzed: the physical and logical resources which group the infrastructure view of the service that consists of considering equipments and devices; and the functional views that can be defined by abstraction of equipments and devices. Those two cases describe functions that must be performed by the next generation services.

In [YLGF08] a prototype of an autonomic service management framework is presented and integrates the Service-Oriented Architecture (SOA), Application-Oriented Networking (AON), and autonomic computing to facilitate the next generation network management. In the autonomic service management framework, a business or management application with end-to-end QoS requirement can be automatically composed from standard service components in a distributed manner, and managed according to service level agreements (SLA) as well. Therefore, to follow the SLA agreements as a whole becomes a difficult task due to the limitation of QoS and management solutions adopted in this work.

In [DAS09], an autonomic network management architecture called ANEMA was presented and the goal is to introduce different mechanisms to allow IP-based networks to manage themselves. To demonstrate the ANEMA architectural capacity, the authors explained how it should be instantiated in a multiservice IP network and how the proposed utility based analytical models are used. They have also implemented a simulator of the system and performed a set of simulations based on several proposed scenarios. The results obtained have been discussed, and a small-scale testbed composed of several autonomic routers and servers implementing the ANEMA compound was built to improve this contribution. However, to develop acceptable solutions to the operators, it is necessary to implement ANEMA in large scale, studying the effect of local decisions on the entire network and how these decisions are propagated among the autonomic entities.

Based on autonomic computing and virtual network concepts, Tizghadam et al. developed a system called Autonomic Network Control and Management System (AORTA) to cope with the management in a self-organized way [TLG08]. They also developed an algorithm whose main function is to robustly manage the demands from any source destination, having the minimum service interruption at the core. By this way, the algorithm is useful to accommodate new requests for connections along the paths. However, AORTA contains problems in terms of long-loop, survivability, robustness and performance optimization.

In [SBEKAP09] an Autonomic Load Balancing Algorithm (ALBA) is presented to improve the utilization of capacity in WLAN while guaranteeing the QoS of VoIP applications. ALBA relies on an autonomic architecture to cope with unexpected changes in the network topology. ALBA offers autonomic management capabilities to access points that ensure the accommodation of more user traffic and overall network capacity improvement compared to default signal strength based connectivity. The results obtained through simulations show how the micro behaviors implemented in each AP combined into a consistent behavior that evaluates the load outside the congested areas. Channel Utilization Estimation (CUE) metric is used instead of bandwidth to support multi-rate radios. The performance evaluation shows substantial capacity gains, if compared to the traditional signal strength based algorithm. Therefore, ALBA needs to integrate other criterion's to load balancing, at control level, such as metrics at level-2 that help the dissemination of QoS information to best neighbors AP's with good quality of signal, for example.

In the last years, several projects approaching the concept of autonomic management in different ways were developed. The 4WARD [Ab10] is an example of envisioned In-Network Management, where management functions come as embedded capabilities within the devices.

Univerself [Uni13] is a project that addressed the complexity of next generation networks aiming to provide a global autonomic management system. A Unified Self-Management Framework was proposed, regardless current and forthcoming architectures, and abstract to any access technology. The Unified Framework is intended to cope with existing technologies and proposing proper self-x and cognitive mechanisms, adopting a cleaned state strategy. It is expected, through the self-x ability potentialities of this novel framework, to raise new opportunities for service providers and to dynamic manage network resources and future internet services.

AWARE project [Awa13] provided a supportive environment for research into self-awareness in autonomic systems. AWARE aims to create computing and communication systems that are able to optimize overall performance and resource usage in response to changing conditions, adapting to both context (such as user behavior) and internal changes (such as topology). This requires breaking through the tradition of fixing abstraction layers at design time, which hides issues at lower layers (e.g., by hiding mobility, heterogeneity, or drops in performance), but inevitably limits the scope for optimizing resource usage and responding to changing conditions. Creating awareness at the autonomic nodes level, it supports dynamic self-expression to adapt the tradeoff between abstraction and optimization.

SAIL project [Sai13] designed technologies taking into account the fact that information and applications are mobile, and can be found in many places inside the network. Thus, end-users need to be able to address content directly, instead of addressing servers to get the closest applications and content around in the network. SAIL leads with the varying demand, and the devices that compose the network need to adapt rapidly to connect applications and end-users, taking advantage of all available resources. SAIL enables the co-existence of legacy and new networks via virtualization of resources and self-management, fully integrating networking with cloud computing. It also embraces heterogeneous media from fiber backbones to wireless access networks, developing new signaling and control interfaces, able to control multiple technologies across multiple aggregation stages.

Self-NET project [Sel11] introduced cognitive self-managed elements based on a hierarchical feedback control cycle. Self-Net management approach relied on a distributed system built of Autonomic Elements (AE) enabled with a cognitive cycle: monitoring, decision making and execution. Monitoring regards to collect the information about the AEs. Decision making includes learning, reconfiguration and situation awareness. Execution is the enforcement of the decisions on the AE, taking advantage of the self-configuration capabilities of each AE.

BIONETS project [Bio09] worked on evolutionary and adaptive mechanisms in networking management and services, to self-organize in a fully autonomic way. It considered peer-to-peer communication paradigm and the service adapts to the surrounding environment, as occurs in the nature.

All these projects addressed self-management mechanisms, some of them distributed in the network, but none of them addressed the communication framework for the nodes to organize, gather and transfer management information in the network.

The next Subsection will give more detail about the In-Network Management (INM) concept developed in 4WARD project, since it is the basis of the work developed in this Thesis.

2.3.2 In-Network Management (INM)

The 4WARD project [DG09] adopted perspectives to create and design a Future Internet architecture in long-term, with the aim of using a clean slate approach. INM [ADM+09, FBP+08, DG09] is a new paradigm studied in the scope of 4WARD project, and supports management functionalities by the means of a fully distributed architecture. The main objective of the INM is to design management functionalities located inside the network elements and services that are involved. Thus, each element has the capability to take decisions based on the knowledge from the other elements, once the INM ability is based on a distributed control mechanism. This mechanism provides continuous interactivity between nodes in order to exchange information about each node (and therefore the network). This information will allow the network to make automatic decisions, reacting to network changes (such as link failures, load variations, etc.) and continuously optimizing the network resources (in both physical and virtual networks, for users and services) according to optimization mechanisms. As it was previously referred, future networks are expected to become even more complex and dynamic. In these scenarios, the management of the networks through the conventional and centralized approaches is extremely difficult and contains serious scalability problems. The role of management in these scenarios may be different due to the dynamic behavior of the entities.

As opposed to traditional management, in INM each entity interacts with its peers with the ability to take decisions based on the knowledge from the other elements [ADM+09] (Figure 2.7). In the traditional network management, the administrator of the network has the central control of management decision, interacting with the network management through the management commands. This approach uses an external server turning out to be inadequate in terms of scalability.

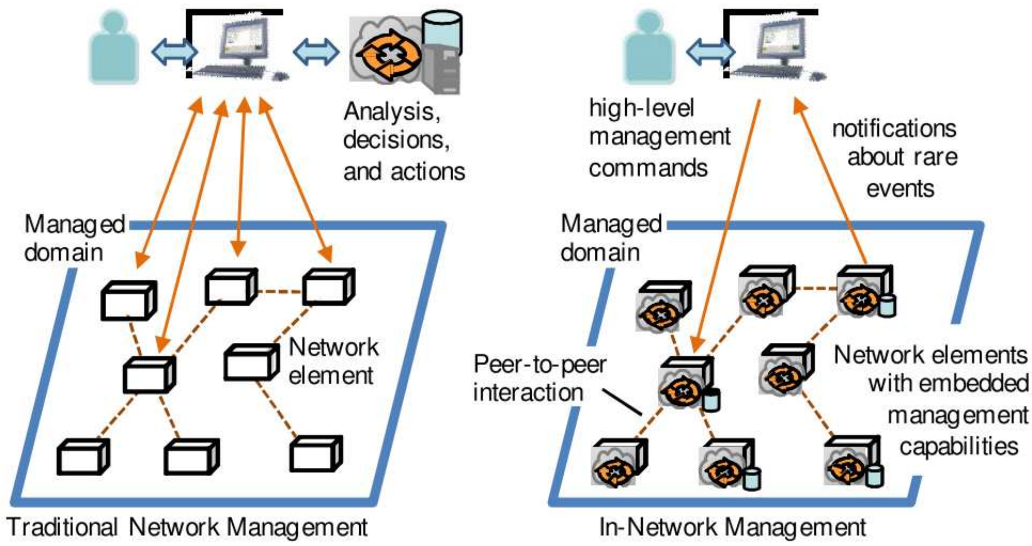

Figure 2.7: Traditional (left side) versus In Network Management (right side) [BDMN09]

Consequently, in In Network Management, for the INM entities to know each other and to communicate to perform totally distributed management decisions in an efficient way, specific functionalities and processes are required to provide the means for the nodes to cooperate. The processes comprise bootstrapping of nodes/network, discovery of nodes, exchange of network information between nodes, dissemination of local decisions from the nodes to other nodes, and final decision enforcement dissemination to perform distributed making decision processes. These mechanisms will provide the support for communication between INM entities.

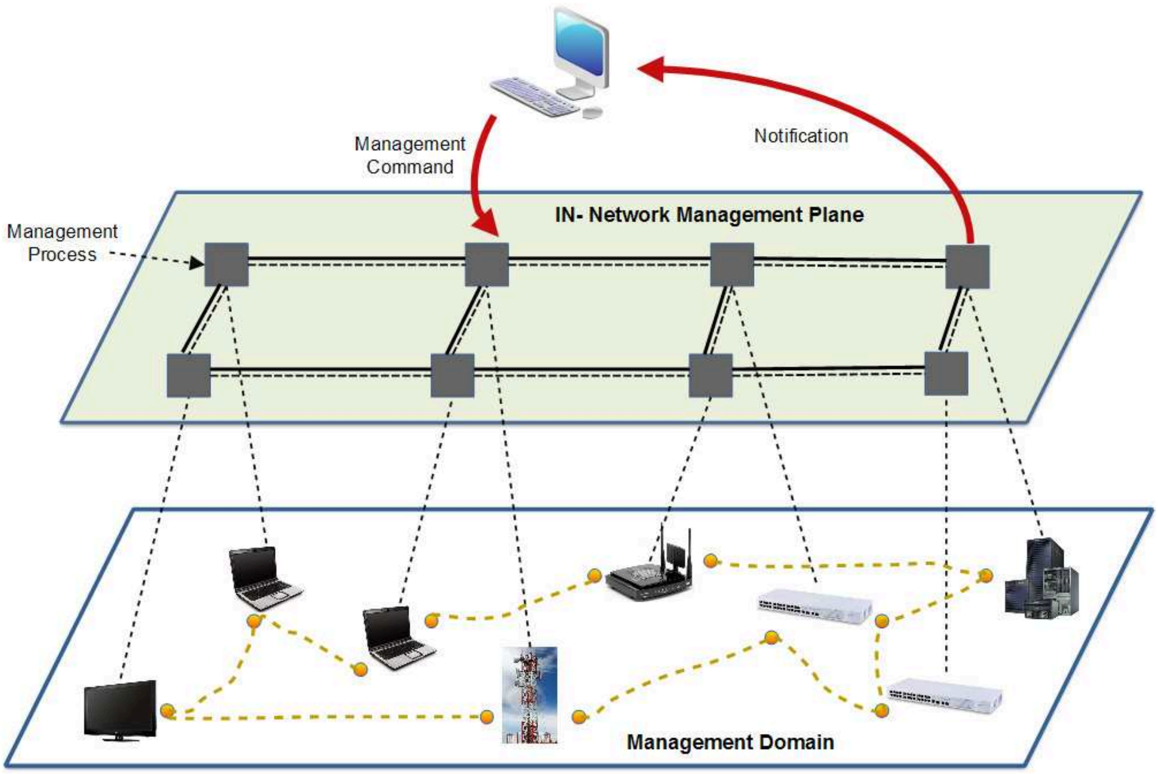

Figure 2.8: In-Network Management Plane, adapted [ADM+09]

Regarding management plane (Figure 2.8), In-Network Management relates to management plane inside the network, not in dedicated and external servers as in the case of the previous mentioned approaches. Moreover, this concept exhibits more autonomic behavior inside the network, through high collaboration between management processes, and the management processes embedded in each network device over a management domain as well. Also, each network management process interacts with the neighborhood through an overlay association. Further, this communication can be done through peer-to-peer interaction and relying on different propagation schemes to enforce management processes that will allow a level of real-time awareness notification and reconfigure in response to node addition or failure. In the following, Table 2.1 shows these functional requirements: we identified some important requirements such as situation awareness, scalability and comprehensiveness of management Information. We base the communication infrastructure proposed in the scope of this Thesis taking some of these requirements into account.

Table 2.1: In-Network Management Functional Requirements (source [DG09, ABCA06])

Requirements	**Description**
Heterogeneity	There must be support for heterogeneity on different dimensions.
Interoperability	INM relies on interoperability between different devices to support management operations.
Real-Time-Support	Support management operations within limited time constraints (i.e. real time, near real time).
Small Footprint	Must have a small footprint with respect to storage space, bandwidth consumption, energy consumption, and other resources.
Level of Localization	The localization of management tasks/control loops to a small subset of components is a prerequisite for scalability and a small footprint.
Situation awareness	There must suitable mechanisms for real-time monitoring of network-wide metrics, group size estimation, bootstrapping, nodes and topology discovery, data search and anomaly detection.
Level of Integration	Management tasks need to be tightly integrated within network components and subsystems and fully embedded.
Adaptability	Must be able to embrace current and proprietary technologies, allowing them to be maintained in existing systems, and allow migration/evolution to new approaches.
Security	Must address security along different dimensions.
Learning	Must provide mechanisms to learn new knowledge about the network.
Scalability	Must support scalability in terms of network size, e.g. the number of network components to be managed. Must provide mechanisms to aggregate the network in domains or multi-domains.
Robustness	Must be robust with respect to failures in components and subsystems.
Comprehensiveness of Management Information	The Information model of INM must be comprehensive in the sense that it must provide sufficient expressiveness, extensibility with respect to future requirements and flexibility. There must be support for efficient dissemination of management information.
Functional Comprehensiveness	Must provide functional richness to support a variety of essential management tasks.
Interactivity	Must provide the possibility for humans to intervene on different levels of abstraction.
Comprehensibility	Must be comprehensible in the sense that it is possible to understand parts of the system, for example, in situations of failure.
Extensibility	Must assure that capabilities of nodes can be extended with new functionalities.

2.4 Mechanims for Nodes Communication

This Section presents existing approaches for nodes communication through bootstrapping, neighbors and topology discovery, nodes association and organization and exchange of management information.

2.4.1 Wireless Communication Network

This Section presents the main features of the control processes and management performed at the MAC layer of the IEEE 802.11 [IEE07]. This information will then be required to understand the mechanisms proposed on top of IEEE 802.11 wireless environments.

2.4.1.1 General Architecture

The IEEE 802.11 protocol is based on an architecture where the system is divided into cells. When each of these cells is controlled by an Access Point (AP), this control is called Basic Service Set (BSS). Otherwise, when there is no central controller or AP, this control is called Independent Basic Service Set (IBSS). The Extendend Service Set (ESS) is the name given to a set of one or more BSS interconnected such that at the logical point of view, all stations could be considered part of the same BSS. The Domain System (DS) is the one who connects the APs in an ESS, allowing communication between BSSs. In terms of identifiers level, the most comprehensive is the Service Set IDentifier (SSID) that matches the name of a particular IEEE 802.11 wireless network. An ESS will necessarily have to maintain a unique SSID; it means that each of the BSS must share the same identifier. This identifier is a string which contains up to 32 characters. The standard also enables the broadcast of information, allowing the creation of virtual access points (Figure 2.9).

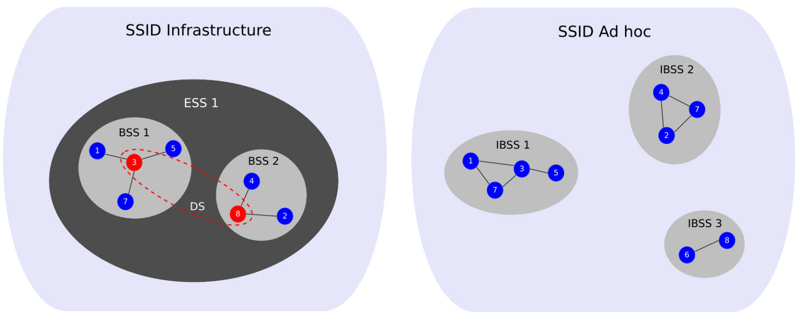

Figure 2.9: General architecture of the IEEE 802.11 standard

In infrastructure network, the Basic Service Set Identifier (BSSID) identifies each BSS, which is the MAC address of the AP. In a IBSS, the BSSID is the MAC address generated locally, containing 46 random bits (the bits are universal/local '1' and individual/group to '0'). A BSSID with all bits to '1' indicates that it is a broadcast, and can only be used in active probing to send Probe Requests signaling packets. The remaining 46 bits are random, generating an unique addresses (Figure 2.10).

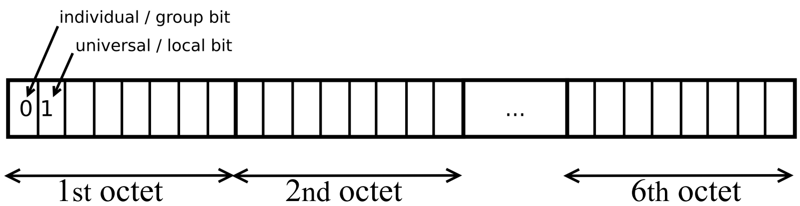

Figure 2.10: Constitution of the BSSID in IBSS under IEEE 802.11 standard

The IEEE 802.11 standard offers two different operation modes: infrastructure and ad-hoc (Figure 2.11). In the first one, there are central management points or APs: the wireless nodes, designated as Stations (STAs) can communicate directly with the APs in their coverage area. In ad-hoc mode, the communication is performed without using any infrastructure or AP, but with directly connections between the various STAs. After being initialized through a bootstrapping phase, each STA will look for others in order to associate/join with them. This association is not definitive, since due to the STAs mobility, they can get out of range of those who are associated. When this happens, it is necessary to trigger a new search process to find a STA that can be associated.

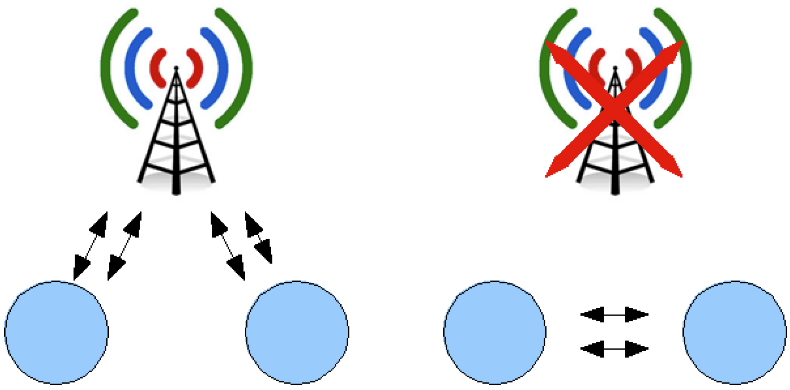

Figure 2.11: Infrastructure vs Ad-hoc

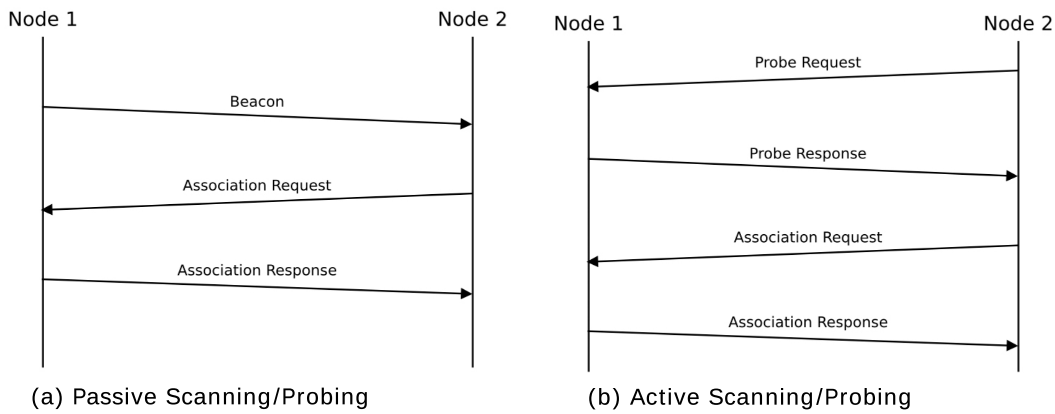

Figure 2.12: Passive (a) and Active (b) Modes

The IEEE 802.11 standard also introduced two methods for network scanning: passive and active (Figure 2.12):

- Passive mode: the STAs will listen the transmission channel and wait for the reception of beacons whose SSID is the same. After receiving this type of packets, the node starts the process of association by sending an Association Request message.
- Active mode: the station, instead of waiting to receive beacons from the other stations, it immediately sends a broadcast Probe Request packet, signaling where the SSID is configured. The answers through Probe Responses will only be received by those belonging to the same SSID.

2.4.1.2 MAC Layer

The IEEE 802.11 supports both physical and Media Access Control (MAC) layers, and the current standard being set a single MAC layer which interacts with three transmission schemes (Figure 2.13): Frequency Hopping Spread Spectrum (FHSS), Direct Sequence Spread Spectrum (DSSS), Orthogonal Frequency Division Multiplexing (OFDM). The IEEE 802.11 also defines several packets to perform the management, control and communication over wireless links. The management packets allow the STAs to establish and maintain their communications, and the sub-types are:

<table>
<tr><td colspan="3">802.2</td><td>Data Link Layer</td></tr>
<tr><td colspan="3">802.11 MAC</td><td>MAC Layer</td></tr>
<tr><td>FH</td><td>DS</td><td>OF</td><td>PHY Layer</td></tr>
</table>

Figure 2.13: IEEE 802.11 Protocol Layers [IEE07]

- Beacon: are generated periodically as a way to announce the presence of a STA and informations contain timestamps, SSIDs, Received Signal Strength Indication (RSSI), maxpower, etc. Moreover, the STA continuously scans for channels in order to detect the reception of new Beacons.

- Probe Request: Probe packets are used in Active Scanning mode, when a STA needs to obtain information about the presence of surrounding STAs in the environment, rather than waiting to send Beacons.

- Probe Response: it is the response to the previous request, where the information about the STAs is encapsulated, and those are regarded to: capability information, supported data rates, etc.

- Authentication: it is the process by which a STA accepts or rejects the identity of other surrounding STAs. Here, the STA sends a challenge/response authentication message using for example Wired Equivalent Privacy (WEP). The acceptance or rejection of the authentication depends on the correct response.

- Deauthentication: This packet is sent when a STA decides to finalize the communication (e.g. many missed Beacons).

- Association Request: the association is initiated by sending an Association Request packet. The STA that received the Association Request checks the parameters contained in the packet (e.g. supported data rates), and then sends a reply, accepting or rejecting the association.

- Association Response: the response to the requested association can be positive or negative. If the association is granted, the packet will include information concerning the association (e.g., supported data rates).

- Re-Association Request: if a STA looses signal from one AP and it finds a new AP, the STA will request a re-association with the new AP. This will be responsible for forwarding the data that might be queued in the previous AP to be transmitted to the STA.

- Re-Association Response: resembles the Association Response, but in this case relates to the Re-Association request.

- Disassociation: A STA sends this packet to the AP to finalize the association previously established (e.g., battery exhausted). Thus, the AP can free the resources with this association and eliminate this STA from the associations local table.

2.4.1.3 Generic Management Frame Format

The management packets will be exchanged at the MAC level, and the header structure is depicted in the Figure 2.14. Those fields contain information regarding to frame control flags, duration and contention control, MAC address, flags for sequence control, frame body and integrity Cyclic Redundancy Check (CRC). The IEEE 802.11 standard also defines a type-specific data packets, where the information is carried away and encapsulated at the level of Frame Body field, and can be sent by the Information Elements (IEs).

Figure 2.14: Generic Frame Body

IEs are encapsulated into MAC management packets, and contain descriptive information about the STAs, transmission rate, and also fields reserved for specific vendors information. It is common to sent multiple IEs in each management packet, obeying its format and its pre-defined structure, as can be seen in the Figure 2.15. Each IE is assigned as an unique Element ID and the field length specifies the number of octets in the Information field. The common structure of a IE is as follows:

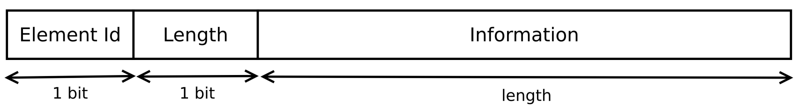

Figure 2.15: Information Element Frame Format

In this Thesis the proposed discovery protocol builds and periodically updates the information about neighbors, which is performed through an extension of the 802.11 MAC signaling process (e.g. Beacons, Probes, Associations/Reassociation and Disassociation). A set of new IE are proposed to exchange management information such as, identifier, MAC addresses, BSSID configurations, hardware capabilities and temporal parameters. The IEs are encapsulated both in the management Beacon and Association packets, and the management information is used by the nodes, for example, to decide the best neighbor or group for association in wireless scenarios, according to the proposed mechanisms for nodes discover and association, described in more detail in the Chapter 3.

2.4.2 Network Bootstrapping

Several bootstrapping mechanisms are found in the literature addressing different types of networks, such as peer-to-peer, ad-hoc, wireless mesh and sensors networks. Especially in the peer-to-peer area, there are numerous mechanisms to provide bootstrapping, centralized, and distributed, such as: CAN [DO03], Chord [SMLN+03], Pastry [RD01], Tapestry [ZHS+04] as centralized; Multicast peer-to-peer and Caching [MH08] as hybrid; and Spanning Tree algorithms [MLAV06] and DDNS-Based [KWSW08] as distributed. The major design problem in bootstrapping mechanisms is scalability; these mechanisms should also fulfill the requirements of robustness, security, availability, automated connectivity, expending low cost messages overhead to initial network formation.

The work proposed in [SSC+13a, SSC+13b] described an automatic bootstrapping of switches by using an algorithm where the controller establishes a path through the neighbor switches that are connected to it by the OpenFlow protocol [Fun12]. Thus, the switches have to search and establish a path to the controller (bootstrapping) through the other switches in the network. Through a GUI (Graphical User Interface) placed at the controller, the OpenFlow switch topology gathered during bootstrapping is shown. Therefore, the bootstrapping mechanism proposed is entirely dependent of the controller and, in case of link failures, the controller starts recovery by itself which takes time after the failure detection.

In [TCS13] the author facilitates secure and seamless mobility across heterogeneous networks through authentication based on bootstrapping using Federated Identity systems. The principle behind this work is in leveraging an application layer bootstrapping mechanism to facilitate network access layer credential generation in a seamless and automated fashion, dynamically when a consumer moves to a new network. Therefore, the bootstrapping mechanism proposed in this work is located at a central server, which is disadvantageous in terms of single point of attack and failures.

According to [GZJL09, GG08, MH10], a bootstrapping mechanism aims to detect and contact peer nodes in the network. In [MH10], the author implements a fully distributed bootstrapping mechanism. Simulation results verify the feasibility of the algorithm for use in practical peer-to-peer mobile ad-hoc networks. The overhead of messages exchange is a disadvantage of this approach.

In [CJSR08], the authors emphasize that, when an autonomic manager is attached into a running system, it is required to have a bootstrapping mechanism. The authors propose a bootstrapping state mechanism that provides a trade-off between speed, state information acquisition, and loading in management system. Therefore, the bootstrapping in autonomic manager should have current and detailed state information, and needs to minimize loading and element/systems management that provides the state information of Autonomic Manager (AM).

In [KWSW07], the authors propose an automated bootstrapping mechanism based on Dynamic Domain Name System (DDNS). This approach detects one existing peer-to-peer infrastructure and automatically joins the peer nodes into the network. The authors emphasize that the goal of the bootstrapping mechanism is to find an already existent member of peer-to-peer system. Otherwise, without bootstrapping process, the peer-to-peer networks will work in an isolated way, reducing the overall system performance for all peers in the network. In addition, the authors describe a system model that consists in a set of peers that are connected by a common communication network, like the Internet. In this system model, the peers are reliably sending messages, and the peers join and leave the network without sending any further message. However, this approach has limitations when a DDNS fails.

In [MLAV06], the authors propose a self-organized bootstrapping mechanism for directional wireless networks. The authors propose a bootstrapping model that contains three main requirements: (i) a distributed bottom-up algorithm that builds a spanning tree, (ii) a resource discovery algorithm for efficient dissemination of local connectivity information, and (iii) a synchronization protocol that guarantees the connectivity from local interactions. In addition, the authors describe that a complete bootstrapping solution requires the integration of algorithms and protocols for determining connections, exchanging information between nodes and guaranteeing coordination and synchronization. Therefore, the main challenge is that the nodes need to have local connectivity information and a limited number of transceivers.

In [MH08], the authors address the bootstrapping problem in mobile ad-hoc networks. To solve this problem, the authors present a method of bootstrapping in peer-to-peer overlay networks running in ad-hoc networks environment. This method involves multicast peer-to-peer overlay join queries and responses, and caches the results from all nodes. In addition, when one node desires to join a peer-to-peer overlay network, this node multicasts a network-wide join request. The main idea in this work is to apply the peer-to-peer techniques into mobile ad-hoc networks. Therefore, the limitations found (accurate peer information dissemination, local/global synchronization and election of the best super-peers) in peer-to-peer overlay networks are also inherited from ad-hoc networks. Table 2.2 summarizes the common characteristics of some described approaches.

Table 2.2: Comparison between some bootstrapping mechanisms

Works	Network Type	Mechanism	Scalability	Robustness	Distributed
[CJSR08]	Integrated Digital Enhanced Network (iDEN)	Bootstrapping state	no	moderate	Yes
[KWSW07]	Overlays	Peer Cache and Mediator-Based	moderate	moderate	No
[MH08]	MANET's	Multicast P2P and Caching	moderate	high	Hybrid
[MLAV06]	Directional Wireless Networks	Spanning tree algorithms	high	high	Yes

According to the limitation of this literature review, we developed a scalable and portable Bootstrapping approach that works together with the HIde and SeeK Directional Decision (HISK2D) [GGS+13, GGP+11, GVMS10] discovery mechanism in order to initiate and identify nodes at different depths. To make it possible, several steps have to be designed, including the automatic bootstrapping of each node. The main goal is to achieve a distributed network management, i.e, which does not depend on external servers entities. Thus, a single node (e.g. low resource router) has the possibility to initiate the entire network, providing a fully-distributed network in both wired and wireless scenarios (more detail in the Section 3.2 from Chapter 3).

2.4.3 Nodes and Topology Discovery

Discovery can be described as a process where each node becomes aware of its surrounding neighbors presence. This process includes assessing quality of links/signals and providing information to identify the better path to the destination. In the last few years, a significant amount of approaches were proposed for discovering nodes/topology in sensor networks [JLG10, YLWX10, AYB10, WLL10, FNJ10] as well as in ad-hoc networks [CSGW10, PR11, HSB10, MMNK13, FECCP13, SWG+13]; its necessity is directly influenced by dynamic topology of their networks. In the other hand, there are many discovery processes that are performed in the MAC-layer and Link-Layer which discovers the physical topology at 1-hop, for example [RGA+04, KKS09]. The discovery processes using collected information to determine the logical topology at IP-Layer are presented in [LZY07, JMB09]. In terms of nodes/topology discovery, there are many solutions already addressed in the literature; the most relevant are Asynchronous Discovery [BEM07, DC08], Bio-Inspired [NND07, CBP00], Directional Antennas [Vas06], Hybrid-Peer Discovery [LZY07], Probabilistic Discovery [KKS09, Vas06], Beacon Assisted [RGA$^+$04] and Service Discovery [HYY$^+$13]. In general, these techniques send non controlled broadcast messages to all neighbor nodes to obtain information from nodes and topology of the network.

On the wired networks side, routing protocols integrate the discovery functionalities in order to create a topology list. OSPF [Moy98] is an adaptive routing protocol that uses Link State Advertisements (LSAs) packets for neighbor discovery. Regarding to commercial discovery approaches, Cisco Discovery Protocol (CDP) [iCS12] is used to show information about the interfaces of routers, bridges, access servers, and switches, and it is a protocol-independent which runs on all Cisco-manufactured equipment. CDP process is made through multicast, and the nodes send periodically Hello packets in order to collect information about the neighbors. However, the information collected is restricted to Cisco manufactured products and it is a proprietary solution, absenting the flexibility to implement new functionalities. *Overlook* FING [Coo11] is another commercial example solution for nodes discovery, that uses broadcast ARP-messages to discover the nodes in the network. In this work the authors study Critical Link Disruptor (CLD) and Critical Node Disruptor (CND) optimization problems to identify critical links and nodes in a network whose removals maximally destroy the network's functions. The conclusion of this study provides a comprehensive complexity analysis of CLD and CND on general graphs, and show that they still remain NP-complete even on unit disk graphs and power-law graphs. The complexity to identify in large-scale scenarios, the nodes whose removals maximally destroy the networks functions, is the main challenge in this work.

According to [RGA$^+$04], the authors propose a beacon assisted discovery mechanism for the wireless networks that operate between MAC and network layers. The proposed discovery protocol operates by listening to augmented MAC-Layer beacons from the neighboring radio nodes, and after that, the best of its associations is selected based on specific criteria. In their prototype, the authors concluded that the implementation for an 802.11b network reduce the routing overhead by the means of separated discovery layer. However, in this approach they considered a Forwarding Node (FN) to forward information between other nodes, and that can become a problem in fully distributed scenarios with high degree of mobility.

The problem to determine the neighbors in a wireless network is emphasized in [BEM07], where the authors consider the problem of node discovery in two different layers: physical and medium access. Therefore, they propose a neighbor distributed algorithm, allowing each node at the network to have a complete or partial list of its neighbors. However, the need for a global synchronization of the network is considered a problem in this work.

The Discovery using Spanning Tree Protocol (STP) is presented in [DL05, PHXQ10] and the authors propose a protocol which establishes and maintains a spanning tree connecting the group of mobile wireless nodes in ad-hoc networks. The protocol is defined by contacting the nodes that are surrounded and recording their information in local tables. Those tables are used to establish the better path between nodes. However, in this protocol, the authors consider the cost based in hops showing inefficiency when the links are congested. The protocol also does not ensure the best neighbors at 1-hop association, which is important to bring benefits to the routing protocol at layer-3 when it is ensured at level-2.

The problem of neighbor discovery in static wireless ad-hoc networks is addressed in [Vas06]. At this point, the authors propose two classes of probabilistic neighbor discovery algorithms, Direct-Discovery and Gossip-Based Algorithms, where any node must receive at least one successful transmission from its neighbors in order to discover them. The directed-discovery algorithm operates with at least one successful transition to discover who the neighbors are. The problem of this approach is that the performance degrades if the estimation of network size is inaccurate.

The T-Man is proposed in [JMB09] by a Gossip-based protocol, which builds the overlay topology in a distributed way. The strong points of this protocol are its readiness, robustness, and simplicity; however, this approach needs a large amount of messages to synchronize the network topology. In [KKS09], the authors abstract the problem of nodes discovery in a network with unknown size, using the gossip method [JMB05] and a problem using direct graphs is reinforced. In this case each vertex represents the node participant, and the edge represents the knowledge about other nodes. Following that line, the nodes that participate in the network will send gossip messages from their local information to others. However, overhead of redundant messages can be observed in this approach.

In [LZY07] the authors propose a hybrid peer discovery mechanism based on multicast and directory service that is composed by three components: Directory, Multicast Advertisement and Central Cache. The first is the main component, as it is responsible to provide a directory service containing peer and service list of all peers in the network. The peer list contains information like interface, directory port, local and an external IP address, while the service list contains the service names and indicates the participating peers in the peer list. However, this approach does not work well when a large number of nodes is observed. In this case, the bio-inspired approaches are other alternatives to discover the nodes in a dynamic network. Many approaches are proposed in this field and many examples are based on software agents for the discovery of the topology [CBP00] [NND07]]. In [NND07], the authors proposed an analogy similar to the ants behavior, using the analogous to the ants communication behavior (which uses pheromones) to exchange the information between the nodes in the network. However, this type of approach usually depends on specific platforms or containers that are used to couple agents that make the management.

Table 2.3 summarizes drawbacks of some described approaches.

Table 2.3: Comparison between discovery mechanisms

Works	**Network Type**	**Mechanism**	**Drawbacks**
[KKS09]	Networks of unknown size	Gossip-Based Algorithms	Overhead of redundant messages
[BEM07]	Sensor	Probabilistic Algorithms	Requires global synchronization of the network
[LZY07]	P2P Systems	Hybrid (Multicast and Directory)	Do not work well when the network growing in number of nodes
[VKT05]	Wireless Mesh and Ad-Hoc	Probabilistic Algorithms	The performance degrades if the estimation of network size is inaccurate
[JMB09]	Overlay	Gossip Protocol	This protocol uses a large number of synchronization messages
[NND07]	Ad-Hoc	Analogy of ants behavior	Depends on specific software agents platforms

In contrast with the previous approaches, we propose a low overhead discovery mechanism namely HISK2D [GGS+13, GGP+11, GVMS10] to provide distributed nodes and network topology discovery. The most important features of the HISK2D protocol are the following: (1) entities may change their roles dynamically according to network events; (2) it contains adaptive Hello intervals in accordance to the amount of nodes that enter or leave the network; (3) events and triggers are executed in accordance to the type of the entity contacted; (4) it supports coexistence of IPv4 and IPv6 under Multicast signaling; (5) it performs directional discovery and exchange of information in different levels of depth; (6) it supports collaborative depths in order to reduce the amount of exchanged messages in the network; (7) it is portable to different architectures and devices. The main issues found in the literature are relative to the definition of the management information that needs to be exchanged to optimize network performance, and the way the management information needs to flow. Thus, HISK2D supports management of larger areas and deal with large amount of management information without to include extra costs in the network performance (more detail in the Section 3.2 from Chapter 3).

2.4.4 Exchange of Management Information

The need for propagation of knowledge is fundamental to most mechanisms, i.e., routing, monitoring and management. The dissemination of general information in the network also contains a widely set of approaches in the literature, and some examples can use flooding [CHLS07], the probability-based [DFKS07], MCDS-Based [Meg06], location-based [SC05], epidemic-based [GO07], cluster-based [CMS08] and stochastic game [LWD13]. In those cases, to have a mechanism that distributes management information between the nodes, considering the scope and type of management information is required. Mechanisms are also necessary to distribute local decisions to set of nodes cooperating to disseminate the results of the management process.

In [DDL+02], the authors implemented the policy management dissemination in a hierarchical domain. Each policy is a java object working in a hierarchical domain in order to maintain the policy attributes, while the policy control object is created to coordinate run-time access that disseminates each policy in the network. This approach uses a central coordinator, which is against the scalability requirements. For policy based information dissemination management, [Zhi09] proposes an Information Dissemination Management (IDM) which controls all steps of dissemination and information delivery. Therefore, management of policies in large-scale systems is extremely complex because of the potentially large number of policies that need to be disseminated. In this sense, [DDL+02] proposes an implementation of an integrated toolkit for specification, deployment and dissemination of management policies. Some management dissemination techniques are developed integrating the use of ontologies [Zhi09, ST05, DBS+07] to organize the information and disseminate them between the nodes; even being useful to large information content, the ontology-based approaches need a well-defined semantic classification.

The work presented in [VCPP13] is based on cognitive heuristics, which are functional models of the mental processes, studied in the cognitive psychology eld. They describe the behavior of the brain when decisions have to be taken quickly in spite of incomplete information. In this solution, nodes maintain aggregated information built up from observations of the encountered nodes. The aggregate status and a probabilistic decision process is the basis on which nodes apply cognitive heuristics to decide how to disseminate content items upon meeting with each other. The performance of the proposed solution was evaluated through simulation and compared with other solutions in the literature. Therefore, this work requires testing in real testbeds to really prove its performance under unstable network conditions, such as the case of opportunistic wireless networks.

In [SGFT13], the authors propose distributed cache management in Information-Centric Networks, where distributed managers residing in cache-enabled nodes decide on which information items to cache. However, a high level of cooperation among the managers is required, resulting of very high computational and communication complexity.

Exchange of information through group communications in distributed management systems is theme of research on [Pea99]. The essential aim is to provide a lightweight communication infrastructure to decrease the overhead, reducing the number of extra messages for communication. A framework using IP plus SNMP for group communication is proposed in [Sch96, Pea99]. The main idea of this framework is to develop autonomous SNMP agents using IP multicasting. However, it lacks the flexibility of multicast group re-configuration according to applications and network demand. In [Lea95], a reliable framework for group communication uses the hierarchy of servers and logical timestamps to ensure reliability and correct ordering of the group delivery. This framework uses unicast connections to emulate group communication, limiting significantly the performance robustness and scalability. Another group of communication framework was proposed in [Pea99] [Aea96] which also lacks the multi-domain management support, which is an important requirement posed by NGNs. The previous mentioned frameworks are not able to cope with current demands and support for large-scale management.

In order to provide a lightweight exchange of management information without central controllers, we propose an extension for the HISK2D that uses a Neighbors Eyesight Direction (NED) function [GGS+13] [GCG+13]. The NED is a completely distributed function that enhances the discovery and exchange of information beyond the limit of the directly connected nodes, choosing the best neighbor node to forward the discovery process and exchange of information according to the knowledge depth (e.g. 1, 2, 3 or even more hops) of the network. With the NED, the nodes have the possibility to request how much they need to know about the network in order to, i.e, provide decision making processes (more detail in the Section 3.3 from Chapter 3).

2.4.5 Nodes Association and Organization

In the IEEE 802.11 MAC standard [IEE07], both in the BSS and IBSS, the criteria for nodes' association is usually the RSSI and Signal to Noise Ratio (SNR). However, having higher RSSI and SNR in some cases does not mean higher throughput, since these values depend on the distance between the transmitter and receiver, as well as on the transmission power (TX/RX) [AKET09]. Moreover, this association criteria was proved to be inefficient in terms of load balancing [LMY+11, CMLD10, WNZ+03] and network performance [AKET09]. Most of the techniques for association take into account infrastructure networks, for the AP association [KNK06, LCC+11, XTL+10]. On the other hand, only few works propose the association techniques for fully decentralized mobile networks, such as the case of mobile ad-hoc networks [JGR08]. Most of these association techniques generally work at network level, suggesting extensions for already existent routing protocols, such as the case of Optimized Link State Routing (OLSR) and Dynamic Source Routing (DSR) [SHP+10, BS09]. Some proposed works considered a hybrid solution between infrastructure and ad-hoc mode, where each mobile device simultaneously works as an AP and as a station [CK08, WBHW11]. In [WBHW11] the authors propose Mobile Ad-Hoc Wi-Fi (MA-Fi) that creates a mobile ad-hoc network into IEEE 802.11 infrastructure mode. MA-Fi also outperforms the standard ad-hoc mode communication and offers throughput comparable to normal Wi-Fi, even over multiple hops. The authors in [JZLZ13] propose an enhanced association procedure named Fast Association Process (FAP) in strong mobility. FAP is introduced with new Association_Data request MAC command that increases the association period and provides fast association process in strong mobility over the top of IEEE 802.15.4 protocols. However, IEEE 802.15.4 cannot maintain association in strong mobility, and is limited at short range networks.

Solutions inspired by social networks are mostly found in the field of routing for Delay Tolerant Networks (DTNs) [DXT09, MMDA10, HCY11, DH07], where the main idea is to use metrics to predict the movement of nodes based on the repetition of nodes' contacts. The problem of choosing the best forwarders in DTNs is shown in [FV11], where the authors introduce sociability in routing, a novel routing scheme that selects a subset of optimal forwarders among all the nodes and relies on them for an efficient delivery. However, the metrics proposed in this work take into account only information at network layer, capturing only frequency and type of encounters of nodes. The work proposed in [LDS03] makes use of the observations of the non-randomness of mobility and, to improve routing performance, the authors propose probabilistic routing, PROPHET, which uses history of encounters and transitivity. However, it is not guaranteed that a node with a higher metric will be encountered within reasonable time. In [HCY11] the authors seek to understand the human mobility in terms of social structures. The authors also emphasize that it is possible to detect characteristic properties of social grouping in a decentralized manner from a diverse set of real world traces. However, it is supposed to work with a hierarchical community structure which tends to adapt slowly to changing needs.

In contrast to previous approaches, we propose an association and connection between wireless nodes in communities through social-based metrics [GGSS12b], [GGSS12a]. Those social metrics measure and predict the Friendship of Neighborhood Nodes, Associated Nodes and Community Nodes, so that the connection between the nodes is performed through this set of metrics to optimize their stability and information dissemination. This approach enables the support of decentralized and distributed management in wireless and dynamic environments: the prediction of mobility and encounters that increase the stability of associations and reduce the effort required for management information exchange between the wireless nodes (more detail in the Section 3.4 from Chapter 3).

2.5 Problems and Challenges

Regarding the traditional management approaches, presented in Section 2.2, centralized groups bare the problem of scalability; weakly distributed approaches, even solving a part of the scalability problem, they underperform in cases of increased number of Agent Managers. On the other hand, strongly distributed approaches contain better performance than weakly distributed approaches, but they underperform in terms of overhead for task migration to other agents. Finally, the absence of a central manager became an important advantage in cooperative management approaches, but the high cost per device to collect, store and act seems to be the main disadvantage of cooperative approaches.

Therefore, the high degree of intelligence makes these approaches more complex to implement, reinforcing the relevance of an extra overhead to migrate tasks to another Agent Manager. Nowadays, most projects and initiatives are envisioning the autonomic and self-management as an alternative to manage complex and dynamic environments. In Section 2.3 we investigated relevant approaches in the field of autonomic and self-management systems. We identified their characteristics, strengths and drawbacks, and assessed their performance in the several phases of a communication infrastructure. The main challenge lies in the fact that the management approaches need to adapt to the constant evolution of the networks, and overcome the above mentioned limitations for autonomic and self-management approaches. Additionally, most current approaches for network management [FKALG+06], [AutaN], [Aut08] take into account the communication between the network elements, and control of the network service layers through ontologies and building blocks developed at a higher layers, thereby omitting to optimize the communication at lower layers.

After careful analysis of the related state of the art works on mechanism for nodes communication as presented in Section 2.4, we argue that most approaches use non-controlled flooding to perform discovery and dissemination of information, which is less efficient with the increasing number of network nodes. Furthermore, most of the works miss the importance of bootstrapping as the initial warm-up of the network or specific new entity; and, according to the Table 2.2, only [MLAV06] fulfills the main requirements of bootstrapping, such as scalability and robustness in fully distributed way. In short, new approaches are necessary to efficiently bootstrap, discover and disseminate information in the network - aiming at solving the existing drawbacks as presented in Table 2.3.

In response to that, we propose a lightweight solution to perform discovery through Hide and Seek concept as well as to exchange of management information through a NED function, which results in the HISK2D protocol. This enforces cooperation between the nodes at larger management areas, including choosing the best neighbor node to forward the discovery process and exchange of information according to the knowledge depth of the network (further detailed in the Section 3.3 from Chapter 3).

Regarding the nodes association and organization, it is highly desired to provide mechanisms that work in the MAC layer, ensuring a lightweight communication process due to the high amount of information exchanged between the nodes. In fact, working at MAC layer ensures a more efficient and scalable information-oriented service at upper layers. On the other hand, a lightweight infrastructure for communication between nodes on top of multi-network platforms and technologies is a key challenge to support distributed network management. A lightweight communication infrastructure can reduce costs of management control, enabling the possibility to exchange a greater amount of information without extra cost for the network. The communication between the nodes should work across various network topologies and technologies, such as wired and wireless infrastructures.

In this sense, we propose and analyze a mechanism for association and connection between wireless nodes in communities through social-based metrics, in order to group the wireless nodes in communities and perform nodes' association and data dissemination. This enforces a higher stability on the associations and a higher efficiency in the distributed management process (further detailed in the Section 3.4 and 3.5 from Chapter 3).

On the other hand, a common challenge is to maintain the distributed management information constantly updated, without compromising the network performance. Nodes can change their roles over time; a node can be a router, gateway or access node. On the other hand, the network operator includes and excludes users in the network, adds and removes services, moving from one place to another or making ad-hoc connections. Network nodes are always susceptible to changes in load requirements and requests for QoS, as nodes or links can fail due to hardware problems, loss of power, cables, frequency interference, misconfiguration or a malicious attack, etc. The network is expected to be collectively aware of the changes in traffic load and of the failure of an entity (a node, a link or both). To optimize the overall performance of the network nodes, a network must adapt itself to constant traffic and connections changes. Due to the importance of the situation awareness, highlighting the aspect of mobility, dissemination support in wireless multi-hop environments needs to be considered as well.

In response to those challenges, we unify the distributed network management integrating management functionalities into the managed devices over distributed multi-network technologies testbed. Thus, each device has the possibility to execute common network management functionalities where each technology has its own peculiarities, becoming a challenge for conventional management approaches (further detailed in the Section 3.6 from Chapter 3).

Chapter 3

Communication Framework for Distributed Network Management

3.1 Introduction

Nowadays, different networks and technologies interact in a global network of networks, shortening barriers between so-called NGNs and today's legacy networks. New access technologies are constantly being introduced in the market, whereas the type and complexity of network nodes can vary significantly (e.g. sensors versus servers/mobile routers). The role and capabilities of such nodes depend on their hardware characteristics. Node cooperation is a key part of this interaction, requiring periodic exchange of information. To increase scalability of the network information exchange, network nodes need to cooperate through an efficient set of rules, policies and criteria, to minimize the time and amount of control messages flowing in the network. The communication infrastructure proposed and developed in this Thesis is endowed with this cooperation between nodes as its main goal.

In distributed management approaches, the servers are not close to all network nodes, especially in large scale networks. In fact, the management functions need to interact with the devices, and share local parameters to optimize future network decisions. Management of large scale networks is an essential requirement, becoming a challenging situation from the operator's point of view. Current distributed management solutions use traditional protocols to provide initial communication, exchange of management information and network organization, often with difficulties to meet the requirements of low overhead impact and fast information exchange. A set of communication mechanisms, proposed in this research work, allow the nodes to exchange and share information over different scenarios and under stable/unstable network conditions. However, many challenges have to be taken into account when striving towards this communication infrastructure without to the centralizing network elements.

This Chapter presents an overview of the communication mechanisms developed, showcasing the concepts, functionalities and interactions between nodes in order to support the requirements posed by self and distributed management approaches.

At the level of the communication between the nodes (Figure 3.1), the direct interaction is formed between those residing in the corresponding coverage area, without centralized network elements. In fact, the nodes mobility translates into a more dynamic scenario, implying greater complexity at the embedded node level due to constant changes in the environment. The communication mechanisms are embedded in each network device and each process interacts with the neighborhood through peer-to-peer interaction, relying on different propagation schemes to enforce management processes, and allow real-time notification and reconfiguration

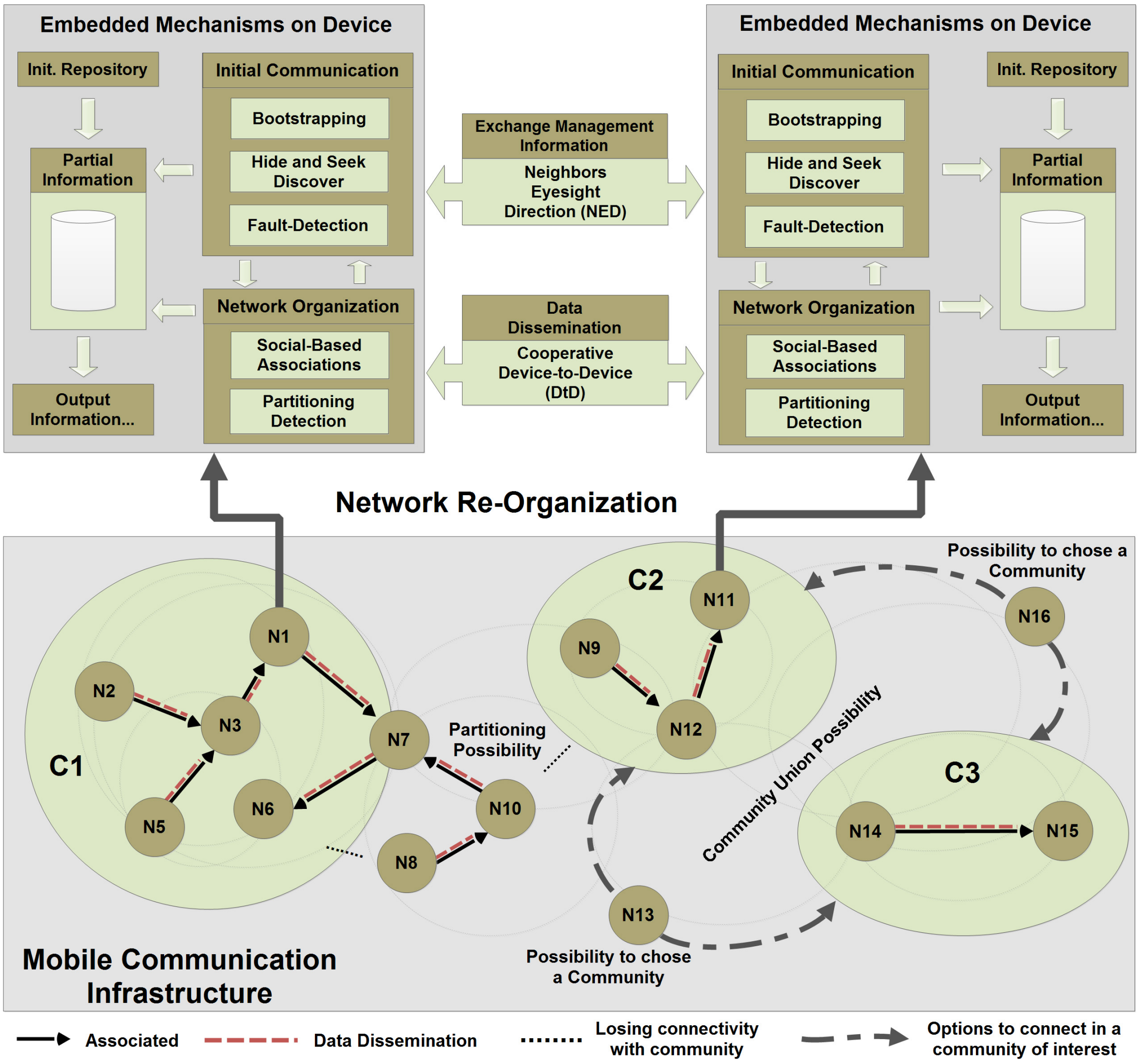

Figure 3.1: Overview of the Communication Framework

in response to node addition or failure.

Regarding the partial information, due to the dynamic nature of the nodes, to maintain the complete information of the network is always a complex task, and often impractical depending on the environment or the scenario. This requires considerable effort from all devices in order to maintain and to share the information, and can even become a disadvantage in terms of information scalability and network overhead. Therefore, keeping partial network information brings benefits to the communication process, enabling mechanisms to control it without adding any extra cost. Moreover, the output information generated from the local repository of each device can be used as the input for many applications, e.g., management tools, autonomic path decisions, routing, etc.

In the following paragraphs, it is described the several stages of the communication framework.

Initial Communication: to enable initial communication, the mechanisms of bootstrapping and discovery were proposed. Bootstrapping is the initialization process of each node, which is formed by several call functions that configure different essential parameters in accordance with the existence of that particular node in the network, so that it operates using the proper protocol. As an example, the configuration of the unique identifier, MAC addresses, BSSID configurations, knowledge of their hardware capabilities, refresh and temporal parameters, etc. Upon this stage, the node is ready to start its activity within the network. Discovery is the mechanism that allows a node to find other nodes in the network, which allows the creation of association links, and enables the communication. The discovery can be performed by direct contact (information collected directly from the neighboring nodes) or indirect (information transmitted/gathered by multi-hop nodes). Thus, when a node receives a message, it can obtain knowledge of all nodes that know this particular information, belonging to a communication group. This is an ongoing process, as in a mobility scenario the conditions will vary over time. Thus, HIde and SeeK Directional Discovery (HISK2D) mechanism is proposed, comprising several stages of initial communication, including bootstrapping, discovery of nodes and topology, and fault-detection.

Exchange of Management Information: this phase corresponds to the indirect contact between nodes, i.e., the exchange of information through multiple relay nodes until reaching the depth (in number of hops) limit. The mechanisms for knowledge sharing in distributed communication aim at increasing the efficiency and speed of convergence of information sharing. This knowledge is mainly related to the nodes at a given moment, belonging to a community and possessing characteristics associated with the community. For this reason it is necessary to take into account the high degree of dynamic scenarios, in which nodes can join or leave a community at any time. This cooperation is provided by a Neighbors Eyesight Direction (NED) function, that enhances the discovery and the exchange of information beyond the limit of the directly connected nodes. The benefits provided by this directionality include choosing the best candidate among the neighbor nodes for the discovery process, and the exchange of management information according to the knowledge depth of the network (e.g. 1, 2, 3 or even more hops). The role of depth is to ensure the possibility to define the degree of knowledge of a node about its network neighborhood. To monitor and detect failures (nodes or links), a fault detection functionality can maintain the consistence of information management between the nodes whenever possible, especially on top of unstable network, as it is the case with wireless networks. The fault handling is divided into critical and warning levels. The critical failures that affect devices, links and the network, involving device crashes or link failures, should be handled reactively. Therefore, warning level serves for feedback on the status of a device (e.g. CPU and memory available).

Network (Re) Organization having the network partitioned into self-organized communities, each device only needs to have information about the community in which it resides, which helps decreasing the amount of information flowing in the network. However, the partitioning cannot isolate sectors of the network, since communication and cooperation need to exist not only inside of the communities but also between them. This can be accomplished through the nodes in each community; communities also have the possibility of merging into a larger community. Moreover, association loops can occur in the communities, and this implies that a node may report something that is not updated at a given time instant. These time instants are sufficient to cause temporary, or even permanent loops in the network due to misleading information. In order to deal with those requirements, we propose and evaluate how social-based association model improves the node associations and network (re) organization models. By the definition of new social metrics, a mechanism to group nodes into communities according to social interaction metrics at MAC layer with the aim to improve their association and message exchange is proposed. The partitioning and association loop problems have been addressed, enabling the devices to create their own communities, keeping their information automatically synchronized and achieving loop-free operation. These social metrics measure and predict the friendship of neighborhood nodes, associated nodes and community of nodes, so that the connection between the nodes can be established with the objective to improve the communities quality aggregations by reducing the network time, utilization and nodes reassociations. Additionally, the understanding of friendship criteria is quantified by the relation/interaction between the nodes in mobile network infrastructure.

Data Dissemination To enable data dissemination, specifically, multimedia streaming content over mobile networks, the communication between the devices is performed hop-by-hop until reaching the destination devices. However, in case of dissemination of multimedia streaming, the challenge increases due to the size and volume of data content, and the mechanisms for dissemination need to automatically ensure that all multimedia content will be disseminated for the largest number of devices as possible (e.g until the last-hop devices), with low end-to-end delay and maximizing success delivery rates. We propose an approach to disseminate multimedia streaming, where technical (provided by social-based metrics) and social network behaviors are jointly identified, classified and evaluated for infrastructureless network association. The technical behavior is inferred by the devices at the MAC layer, such as the number of neighbor devices, available resources, connections quality and stability, and the number of devices interconnected. These parameters are weighted by technical social-based metrics and user's social network accounts (e.g. Facebook, Google+, Myspace, Twitter, etc), allowing the inference of the cooperative DtD behavior. We use Genetic Algorithm (GA) to optimize the input combination of the technical weights leading to the best solution. Merging technical and social network criteria allows the possibility to maintain the best nodes affinities and associations, enhancing the dissemination of multimedia streaming process.

As already mentioned, the Figure 3.1, *Initial Communication*, *Exchange Management Information*, *Network (Re) Organization* and *Data Dissemination* contain the main research topics, where most of the proposed communication mechanisms were developed and analyzed so far. Moreover, a proof-of-concept through real testbed is performed in order to prove the feasibility of our mechanisms. The HISK2D with NED function is part of the *Initial Communication* and *Exchange of Management Information*, and it is implemented as a prototype written in C/C++ (further detailed in Section 3.3.1) to be evaluated in the real testbed under heterogeneous networking environments and unstable conditions. The mechanisms that comprehend the stages of *Network (Re) Organization* and *Data Dissemination* were implemented and evaluated through Network Simulator v.3.9 (NS-3) [NS-10].

These research topics are described in more detail in Sections (3.2, 3.3, 3.4 and 3.5). The Section 3.6 presents the proof-of-concept of the mechanisms HISK2D with NED of the communication framework in a multi-network technologies testbed.

3.2 Initial Communication

This Section is mainly supported by the Papers I and II presented in the Table 1.1 from Chapter 1 and part of the information from Paper III presented in the Annex A.

The first step in the communication between nodes is the nodes initialization through a bootstrapping process. From a single node viewpoint, bootstrap means acquiring the initial information, e.g. setting identifier for the nodes, call the functions (i.e. discovery function) or set initial management policy strategies. At this point, the bootstrap mechanism works together with the discovery mechanism to acquire the initial information about the neighbours of a specific node (Figure 3.2).

To start the communication between the nodes, a discovery process can be initialized by the bootstrapping process, discovering the surrounding nodes and also maintaining the information updated (including the network status and events). It is required to exchange

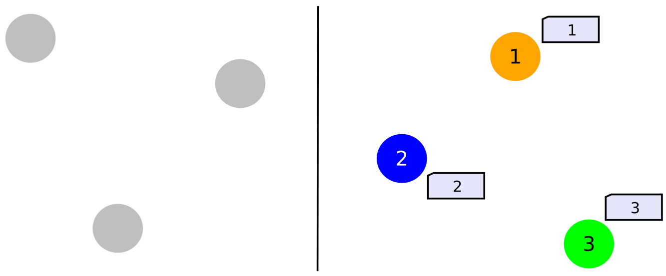

Figure 3.2: Boostrapping generic process representation

initial information to acquire the primary contact information. Discovery can be described as a process in which each node becomes aware of its surrounding neighbors presence. This process includes assessing quality of links/signals and providing information to identify the better path to the destination. The discovery process can work in different network layers, which discovers the physical or overlay topologies. The discovery process can use collected information to determine the logical topology. As depicted in the Figure 3.3, it shows a generic discovery process, where each node collects and sends information about its surrounding neighbors in the same coverage area as well.

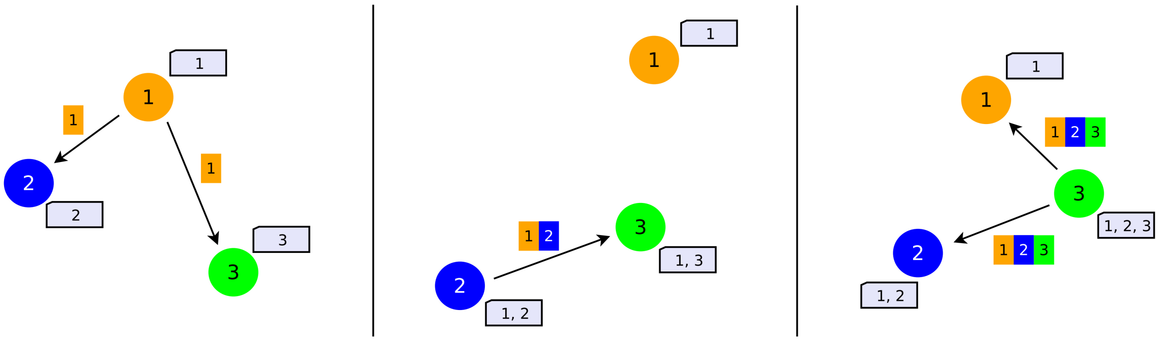

Figure 3.3: Nodes Discover generic process representation

As aforementioned in the Section 2 of state-of-the-art, bootstrapping and discovery of nodes/topology are issues already addressed in the literature. However, in most of the proposed mechanisms, the nodes send uncontrolled broadcasting messages to all neighbor nodes to obtain information from the topology of the network. The major challenge in the design of a bootstrapping and discovery mechanism is to ensure scalability and robustness, which becomes more complex and less efficient with the increasing number of nodes entering and leaving the network. In order to overcome this challenge, bootstrapping and discovery mechanisms should work together in order to discover directly connected nodes to get their management information, roles and optimize how they should operate.

To deal with the requirements presented, HISK2D is proposed, which is a new protocol for bootstrapping, network discovery, and information propagation and synchronization between the nodes. The HISK2D is designed to optimize the initial communication of relevant and sufficient information on each node. In the proposed bootstrapping process, every node starts the discovery of its neighbor nodes, creating a collaboration group of nodes or joining with an existing one. Our bootstrapping and discovery algorithm is based on the *Hide and Seek* game analogy, where the main goal is to find all hidden players in a specific area. Bootstrapping and discovery have strong correlation: when a new node enters in the communication infrastructure, the bootstrapping process configures initial information (e.g identifiers, timers, local repository functions, etc). After that, the neighbor discovery is started.

Two types of nodes are considered in the HISK2D protocol: seeker and hider. This means that the seeker has the specific function to seek other nodes in the network, and the hider is a hidden node at the moment. A seeker node sends directional contact messages to the directly connected nodes. Once contacted, the hiders and seekers synchronize their knowledge, keeping track inside of the local repository of each other, which is denoted as partial view. When being contacted, the hider becomes a new seeker node and the process is repeated until all nodes have been contacted. With regard to the complexity avoidance of constructing and managing the nodes, we develop all interactions between bootstrapping and discovery based on an automated process. This process creates, exchanges and sets up the nodes dynamically without involvement of the administrators. This idea significantly facilitates the administration of the group communication infrastructure.

HISK2D avoids extra overhead of synchronization messages due to the cooperation and collaboration between the seekers. The partial view is a local table that records initial information, in terms of identifiers (MAC and Internal identifier), source and destination IP addresses and roles (seeker or hider).

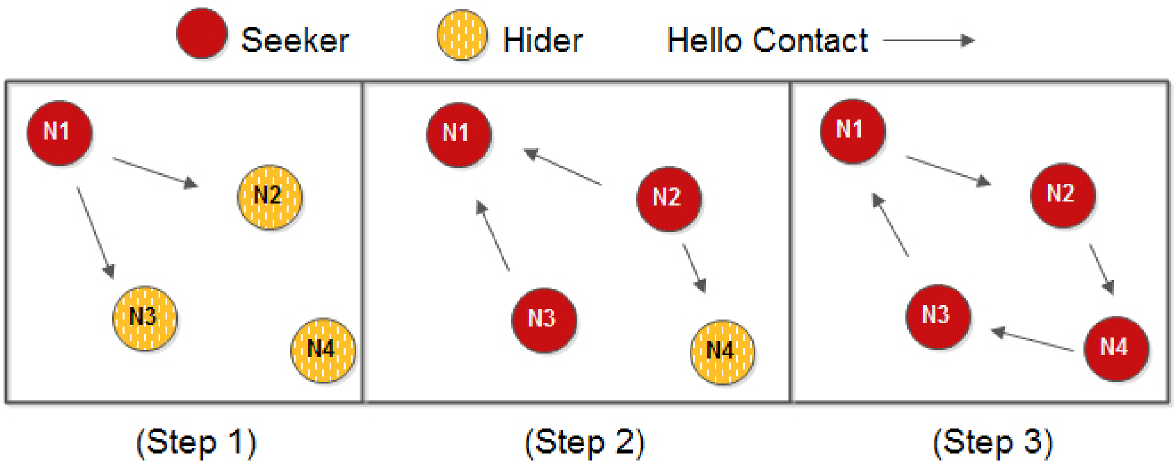

Figure 3.4: Discovery role interaction in three steps

In HISK2D the nodes exchange HELLO messages using at 1-hop depth (Figure 3.4), initially to avoid long cycle messages. Notice that we consider that each node does not need to know the entire network at the beginning. The HELLO interval is adaptive and can increase or decrease according to the number of nodes that are present in each Seeker partial view, and is given by:

$$\mathcal{H}_i = \frac{(\mathscr{L} \bmod (\mathscr{U} * \mathscr{P}_{i,d}))}{\sqrt{\mathscr{P}_{i,d} * \mathscr{L}}} \tag{3.1}$$

where $\mathscr{L}$ and $\mathscr{U}$ represent the lower and upper time limit previously adjusted as reference (e.g. [10-20]), and $\mathscr{P}_{i,d}$ represents the total number of devices at hop distance (d) in the partial view of a device i. Moreover, ($\mathcal{H}_i$) is the interval time which can increase or decrease according to the total number of discovered devices in the network without crossing the limits $\mathscr{L}$ and $\mathscr{U}$.

Figure 3.5 shows the view of one particular seeker node. Note that, in the beginning of the algorithm, at least one node needs to work as a seeker to start the discovery process.

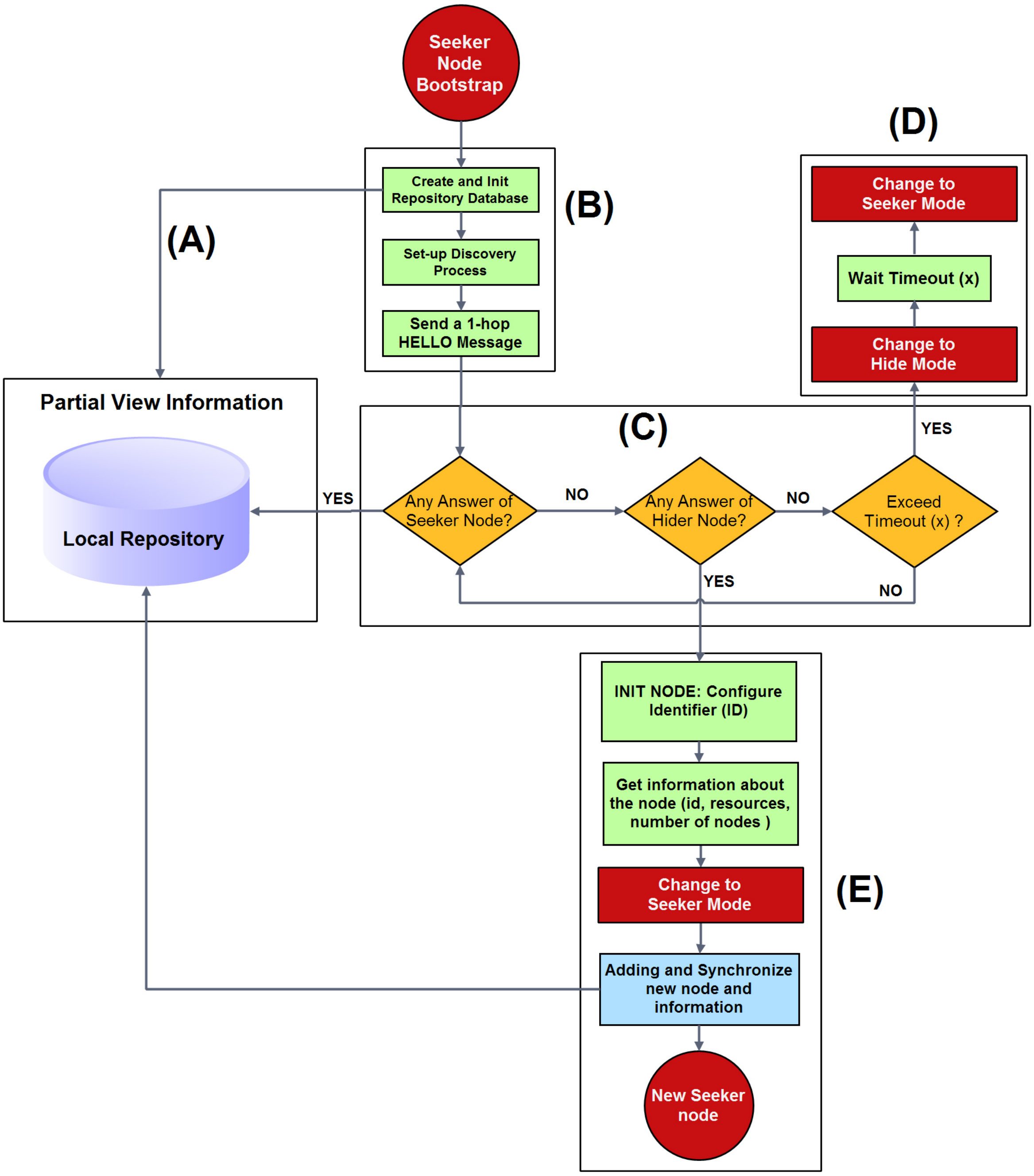

Figure 3.5: HISK2D protocol simple view of the bootstrap and discover sequence

In step A) the seeker node starts the algorithm with the creation of a local repository. Each repository is created locally, and the node is responsible for adding, updating and refreshing all gathered information during the discovery process. The local repository is used to classify the type of information recorded, e.g resources available, network size-awareness, network domain diameters, etc. In the repository, only the partial information is recorded in each seeker repository, which means that, only information about the directly connected neighbors are recorded at the beginning. In analytical terms, the partial view is described with greater detail in the Annex A.

Step B) The Init Repository Database process is responsible to initialize the local repository of each seeker node. It records local nodes information (e.g IP and MAC). The discovery process starts the process of finds surrounding nodes and its information.

Step C) proceeds as a contact between the neighborhood. This contact is realized through messages at 1-hop. Thus, any seeker node can send a message of 1-hop and wait for an answer of any seeker or of any hider (in step (E)). When a seeker node does not receive any contact message before reaching a configured timeout, it executes the step (D). Otherwise, when a seeker node receives a contact of another seeker node, the contacted seeker obtains specific information about resources available, and the new information is synchronized on both seekers repositories. Each seeker node has an internal identifier that performs node differentiation into the network. An internal identifier that controls each node is performed, and it is composed by MAC address plus a random integer number (e.g. 00:45:fa:54:a4-568945). Each node sets this internal identifier in the bootstrapping process.

Step D) works in special cases when the seeker node did not receive any contact under a specific timeout (x). When a seeker node waits a long time without receiving answers from any node, it becomes a hider node. Thus, the probability of another seeker to contact the hider node is higher. To sum up, this process avoids that the seeker nodes spend long time for the hider answers.

Step E) proceeds when a seeker node receives an answer to the contact message sent to a hider node. After receiving the answer, the seeker node gets hider's local information (resources available, etc.), synchronizes the information on the repository, and changes the status of hider to the new seeker into the network. This step is complete when all hider nodes contacted become new seeker nodes. The dynamic on changing the roles of nodes contributes to optimize the contact, spending minimum messages cost as possible in the whole network.

Figure 3.6 demonstrates the automatic signaling process between the nodes, starting with seeker or hider interaction. Note that the signaling process is done by Hello and NodeInfo packets, containing information about the nodes discovered and their local resources.

3.2.1 Experimental Results

To evaluate HISK2D, we develop a scenario that is initially set up as a virtual grid testbed containing OpenWrt Bleeding Edge r28129 (Guest Xen paravirtualized) as virtual machines on a HP Proliant server (CentOS-5 kernel Xen). The testbed is created through a Python script which automatically generates the virtual bridges and link connections between the virtual machines. Experiments are carried out for networks in grid topology with 4x4 (16 machines), 6x6 (36 machines), 8x8 (64 machines) and also 9x9 (81 machines). In the results, we have considered as evaluation metrics: (i) convergence time, which quantifies the time to discover all direct connected nodes (i.e. lower is better); and (ii) percentage of overhead in bytes, which quantifies the impact of the management control packets in the network, according to the total number of bytes in the network as reference (i.e. lower is better).

The values presented in the graphs below are an average of 5 repetitions and 95% of confidence interval. To perform the packets capture, TCPdump is the software used in each node interface, with an observation time of 10 minutes. The links capacity is considered to be 1 Mb/s. We compare our approach with other well-known discovery baselines, using open-source versions of OSPF version 2, CDP version 2 and OLSR version 0.6.3.

Moreover, all protocols analyzed, including HISK2D, are configured (HELLO Interval) to send every 10 seconds their contact messages. The use of OLSR as one of the baseline protocols is due to the fact that its discovery concept considers complete information of the network (it can also be used in wired networks). This allows to compare the advantages of

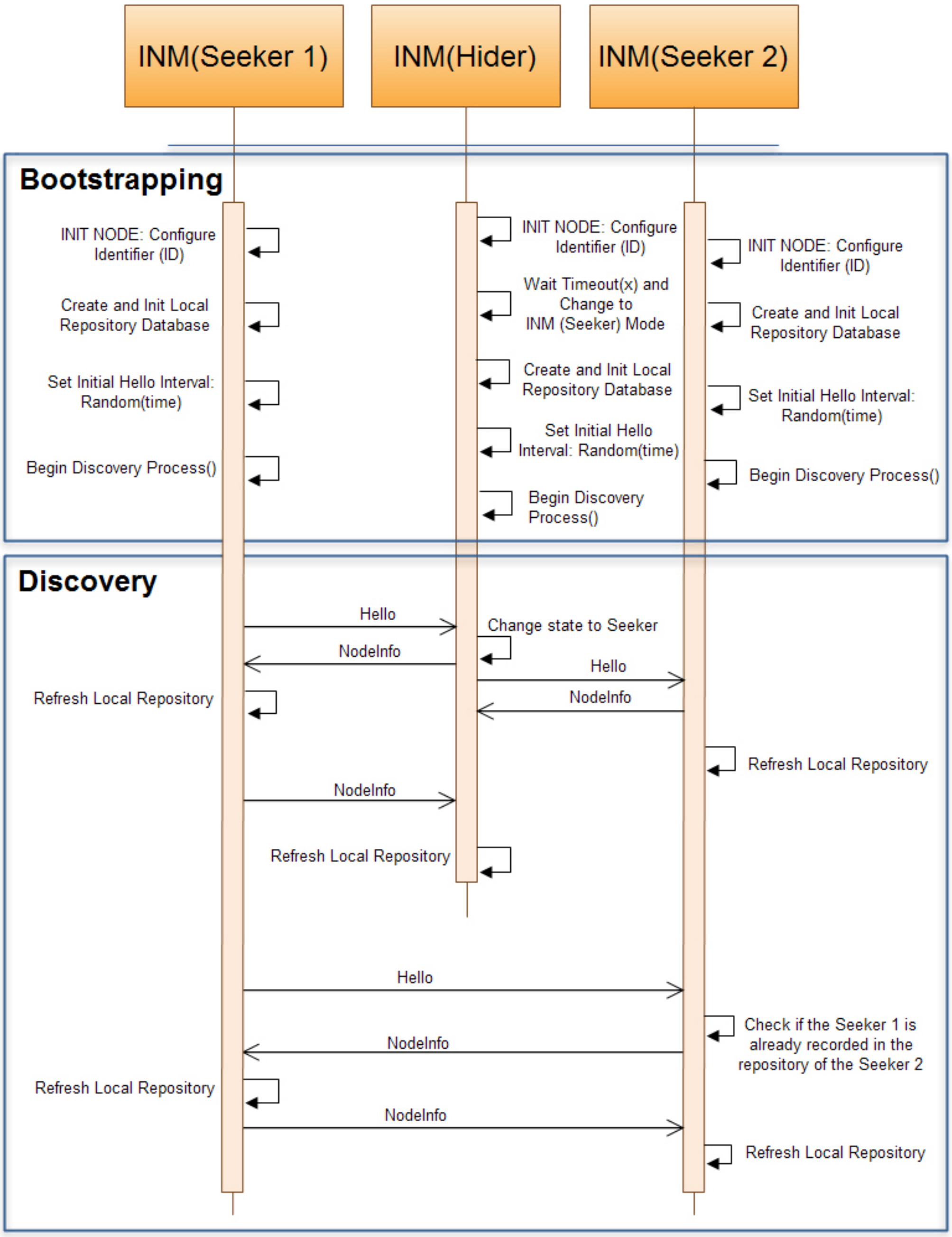

Figure 3.6: Automatic nodes bootstrap and discover signaling process

having the partial or complete information. Moreover, all solutions evaluated (OLSR, CDP and OSPF) are the most used solutions to discovery, and share the same events and conditions regarding the programming techniques.

The purpose of this first scenario is to evaluate the discovery of neighbors at 1 hop by the different protocols. To address this requirement, in OSPF each interface is configured to be in the same area of its directly connected neighbors. CDP is set up with the informations of each virtual machine, and since it can only be used to share information about directly connected equipments, no further configurations are required. Although OLSR, the Multipoint Relay (MPR) coverage is set to 1 hop distance; therefore, the MPR node will be a directly connected neighbor.

Regarding the convergence time, OSPF protocol tends to require more time to discover its neighbors, as shown in Figure 3.7. This was expected, since for large networks, the calculation of Dijkstra's algorithm is complex and consumes significant time.

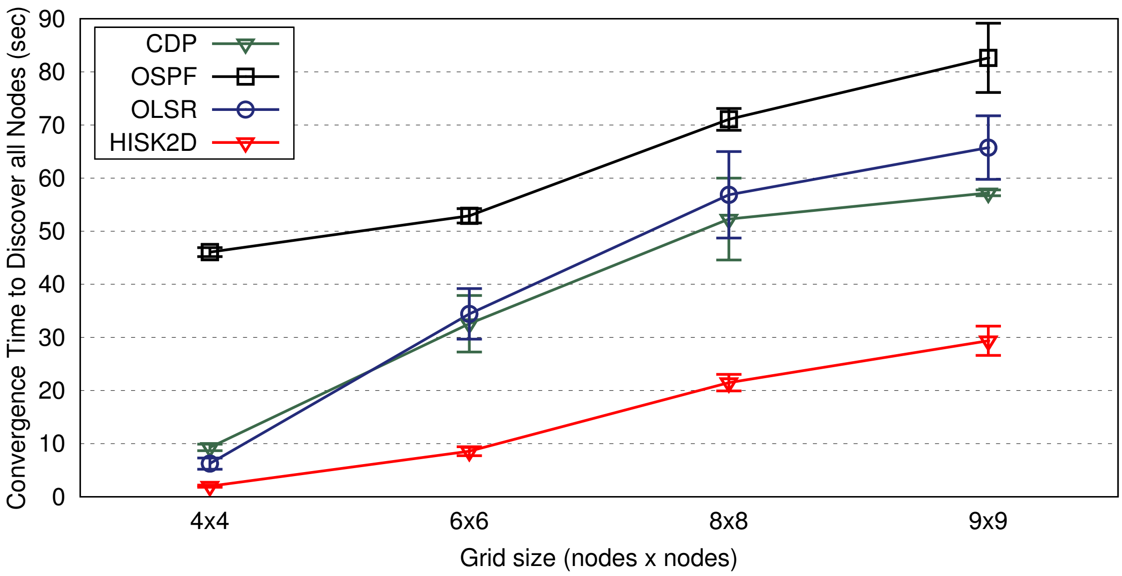

Figure 3.7: Discovery convergence time

CDP and OLSR protocols have convergence times relatively similar, but comparatively to HISK2D, they all require more time to achieve the same result, i.e., using HISK2D each node discovers all connected nodes faster than CDP, OSPF and OLSR. The chosen signaling mechanism, based on request/response, internal triggers and events, is responsible for this fast performance.

In Figure 3.8 it is shown the overhead relative to the control bytes. The traffic measured as reference is a real-time capture performed from the testbed interface eth0. The total bytes used contains messages of router advertisements, ARPs, Pings, DNS request/response and Acknowledgments, resulting in a total of 3103783 bytes.

As observed, HISK2D has a small overhead even in a large network size, requiring less bandwidth when compared to the other protocols. Once more, OLSR highly uses the network to keep the databases synchronized between the nodes due to its flooding mechanism.

In Figure 3.9 it is presented the overhead in bytes, according to the traffic captured through the traffic generator Ostinato [Ost12]. The configuration parameters are shown in Table 3.1. These parameters are set using several streams with different protocols at different rates. The Figure 3.9 presented the curves of CDP, OSPF and HISK2D to highlight the differences between them, regardless OLSR due to the high overhead impact, as presented in the Figure 3.8. It can be noticed that, once more, OSPF and HISK2D are the protocols that less use the network, in contrast to CDP.

The impact of the overhead in a typical operator link is also measured and it is shown in Figure 3.10. Assuming a link capacity of 1 Gbit/s, the percentage of overhead is very low for all protocols, except for OLSR, which has very high overhead impact $\approx$ 5.75e-4%, as presented in Table 3.2. Thus, OLSR not make part of this comparison and more detail regarding to the OLSR evaluations is presented in the Annex A. Once more, Figure 3.10 as it can be noticed, the impact of the CDP and OSPF protocols grow proportionally with the network size.

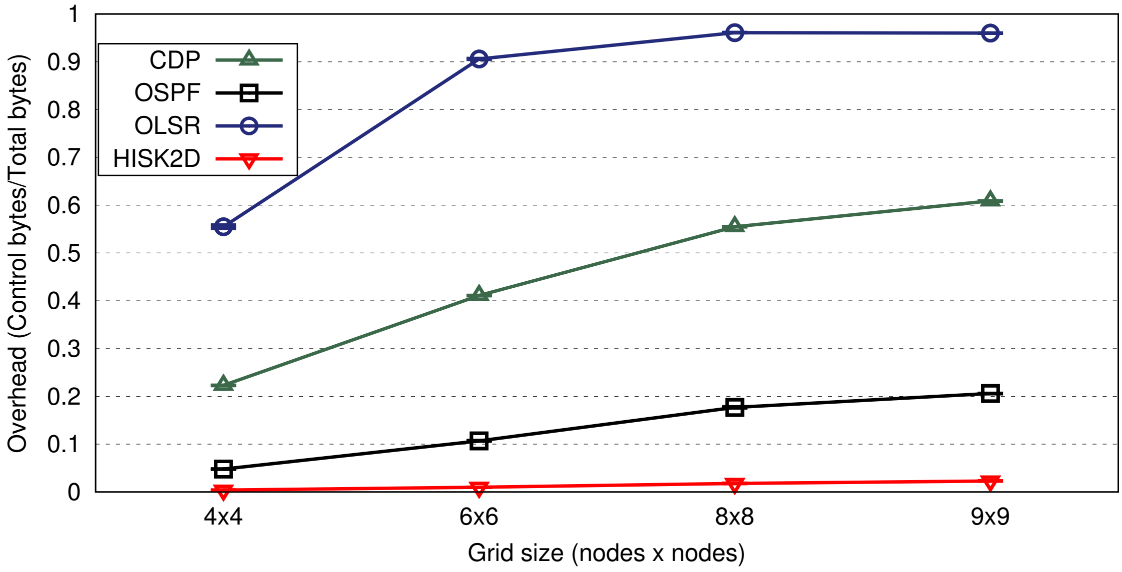

Figure 3.8: Overhead (bytes/bytes) using the testbed traffic as reference

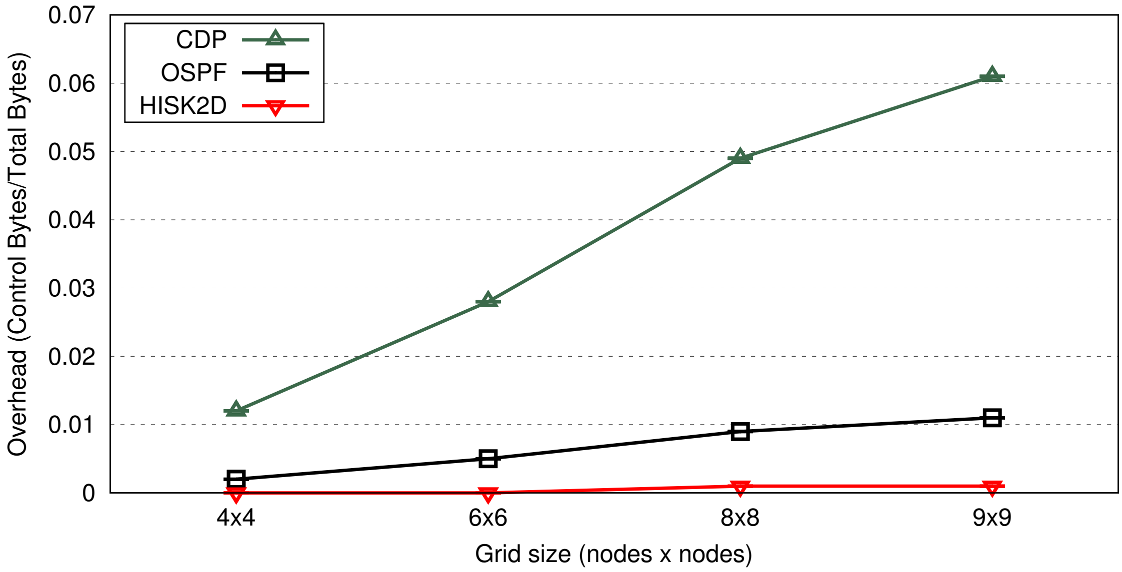

Figure 3.9: Overhead (bytes/bytes) using a traffic generator as reference

Table 3.1: Parameters Configuration

Protocol	**Packet Size (bytes)**	**Packets/s**	**Burst/s**	**Packets/burst**
ICMP	66	-	100	20
IPv6	100-500	-	100	20
IPv4	100-500	-	100	20
ARP	46	50	-	-
802.3 LLC	64	10	-	-
802.3 Raw	64	10	-	-
Ethernet	60	10	-	-

Table 3.2: Average occupancy of each protocol on a 1Gbit/s link, for different networks sizes

Protocol	Network Size	Average occupancy (1Gbit/s link)
CDP	4x4 - 9x9	3.65e-5%
OSPF	4x4 - 9x9	6.21e-6%
OLSR	4x4 - 9x9	5.75e-4%
HISK2D	4x4 - 9x9	5.46e-7%

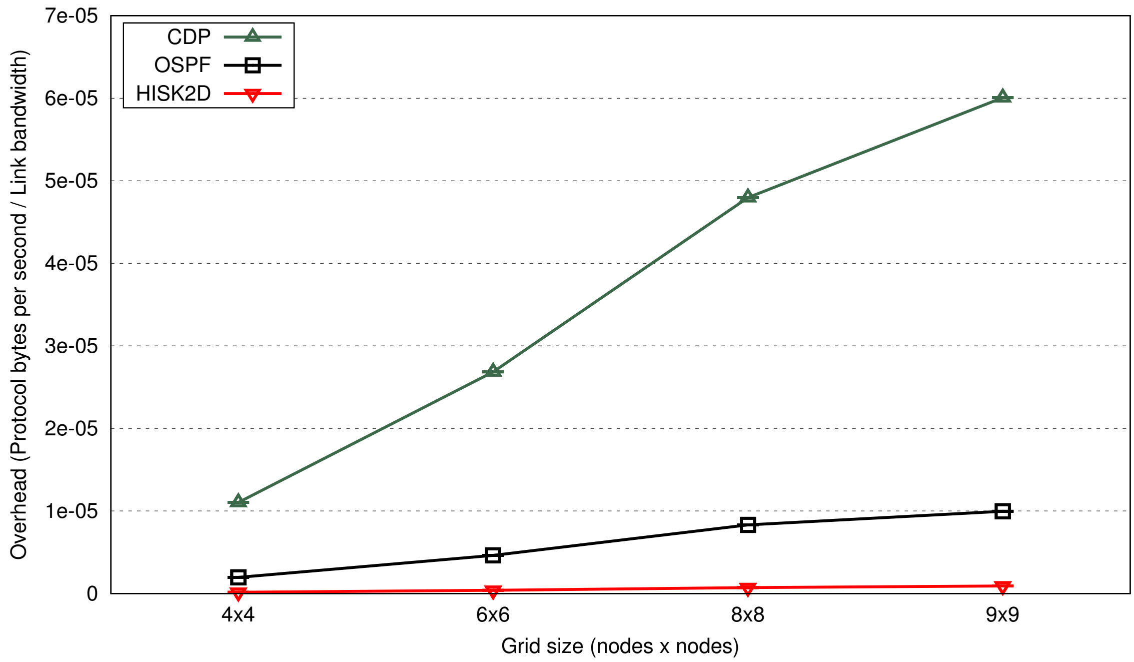

Figure 3.10: Overhead using 1Gbit/s link as reference

3.3 Exchange Management Information

This Section is mainly supported by the Papers III and IV presented in the Table 1.1 from Chapter 1, and both Papers are presented in the Annex A and B.

As already described in the Section 3.2, the HISK2D protocol is proposed to optimize the initial communication between nodes through the discovery process. The interactions in the Hide and Seek discovery are made at 1 hop distance in this initial stage, and they also exchange information about each other, such as identifiers (MAC and Internal control identifier), network status, local resources available (% of free CPU and Memory), source and destination IP addresses. All gathered information is recorded in a local repository of each seeker node. A node in hider mode waits for a seeker contact message (or it waits a random time in case it does not receive any message), and then becomes a new seeker cooperating to discover and exchange information with other seeker or hider nodes. This process is repeated until all nodes have been contacted.

However, the information at 1-hop distance may not be sufficient to support efficiently the distributed management process, since it may benefit from the knowledge of a larger neighborhood. To optimize the amount of information exchanged with low overhead and time at several hops distance, HISK2D uses the NED function, which aims to narrow the directions through which direct nodes or next hop relay nodes will be chosen to continue the discovery and exchange of information. The integration between HISK2D with NED function is also presented in the Annex A.

The NED also brings benefits to distributed management process due the possibility to control the amount of information to be discovered and exchanged in the network, by enabling the information cooperation between nodes. Therefore, this process depends on relay nodes to forward messages until the depth limit is reached. The role of depth (in number of hops) is to ensure the possibility to define how much a node wants to know about the network, and to decide which is the best neighbor to become the next hop relay node in this depth knowledge. The discovery and exchange of network information in HISK2D are complementary processes: initially, the discovery aims to find the nodes according to the requested knowledge of the network in number of hops (e.g. 1, 2, 3...); then, the information gathered through the discovery process is exchanged between the nodes at 1-hop distance, or also through the next hop relay nodes.

The roles played by each node are well-defined: a node can be seeker or hider in the network, according to the interactions performed, as can be seen in the Figure 3.11. The seeker node (e.g. N18) sends messages to its directly connected nodes, according to an adaptive time interval already presented in the Equation 3.1.

The criteria used to choose the next hop neighbor to continue the dissemination process is based on a calculated ranking from the gathered information from neighbors devices. For wireless scenarios, the NED decision is based on the parameters, such as link bandwidth available, network and device's resources available (RTT, CPU and Memory) and by the neighbors devices density. The link bandwidth available parameter induces the NED to find better link conditions to disseminate the information, e.g., congested links or lower link available bandwidth are not suitable to flow the dissemination process.

In case of local and network resources, the NED induces the choice of neighboring devices that have good resources and network conditions to relay the management information, e.g., if a device has high RTT, no free CPU or memory, it is bounded to be a good relay information device. The network devices density parameter induces the NED to choose neighbors that have more density of nodes, e.g., if a neighbor has fewer nodes in its surrounding area, thus, it is not probably a good choice to disseminate the management information to a large number of nodes.

Thus, the NED chooses only the best candidates through eyesight direction, which is given by the set of the following equations.

Link bandwidth available:

$$\mathcal{BA}_i = \sum_{j \in \mathscr{P}_{i,1}} \mathcal{B}_{i,j} \tag{3.2}$$

where $\mathcal{B}_{i,j}$ represents the reference value of the bandwidth available ($\mathcal{B}$) from device i to j. This can be determined by packet pairs/train techniques as suggested by [ML08]. $\mathscr{P}_{i,1}$, is the local information about the neighbors named partial view, i.e., the set of neighbor devices discovered by the device i.

The partial view of device i can be defined by: $\mathscr{P}_{i,d} = \{j : j \in \mathscr{N} \wedge j \neq i \wedge v(i,j) = d\}$, where $v(i,j)$ is a function that returns the hop distance (d) between device i and j if there is a path between them, or 0 otherwise. $\mathscr{N}$ is the complete set of devices in the scenario, where $\mathscr{N} = \{n_i : i = 1, \ldots, N\}$.

Local and network resources:

$$\mathcal{R}_i = \sum_{j \in \mathscr{P}_{i,1}} (\mathcal{C}_j + \mathcal{M}_j) * \mathcal{T}_{i,j} \tag{3.3}$$

where $\mathcal{C}_i$ represents the available CPU, $\mathcal{M}_i$ the available RAM Memory, and $\mathcal{T}_i$ represents the lowest Round-Trip-Time (RTT) measured by device i communicating to device j.

Network devices density:

$$\mathcal{N}_i = max|\mathscr{P}_{j,1}|, \forall j \in \mathscr{P}_{i,1} \wedge j \neq i \tag{3.4}$$

where $max|\mathscr{P}_{j,1}|$ corresponds to the largest Partial View of all connected neighbors, and $|.|$ represents the cardinality of a set.

The global value of the NED is obtained by performing a weighted sum, given by:

$$\mathcal{NED}_i = w_1\mathcal{BA}_i + w_2\mathcal{R}_i + w_3\mathcal{N}_i \tag{3.5}$$

where, w_1, w_2 and w_3 represent the weights of the Equation (3.2), (3.3) and (3.4). The maximum value returned will determine the best eyesight direction neighbor chosen by the device i. Note that each device i will only consider the neighbors $j \in \mathscr{P}_{i,1}$ to determine the NED global value (3.5). Focusing on the decision function (NED) factors, we consider the absolute values for link capacity, RTT and the maximum number of links and interfaces. Moreover, the factors respecting to local node resources, like CPU and RAM memory were normalized, e.g. % of CPU and RAM available. Moreover, We consider the weights for $\mathcal{BA}_i$, $\mathcal{R}_i$ and $\mathcal{N}_i$ with balanced influence of 1/3 for each weight.

In Figure 3.11, by default, *Seeker* nodes have knowledge at depth 1-hop, being also possible to configure a node with depth larger than 1-hop (e.g. N2, N16 and N19). As an example, the initial depth of N19 is 3, and all nodes directly connected to N19 will send their depth and knowledge (partial view) to N19. Then, the NED function will choose a 1-hop neighbor according to the criteria presented in the Equation (3.5) to continue the discovery and exchange of the already obtained information to this chosen node. Then, the best node to continue the discovery is chosen to be N14, and information about N18, N20 and N24 will be exchanged to N14. Node N14 will repeat the 1-hop discovery and will choose a new direction (e.g. N9). At this point, node N14 will have information about discovered nodes N9, N18, N20, N24, N19. Then, the next hop chosen is N9, and information about all nodes discovered at this point, plus the ones discovered from N9 (e.g. N10, N14) will be sent to the chosen node N8, which reached the depth limit. All information collected from N19 to N8 will back track to all nodes that belong to the NED paths (e.g. N9, N14) up to the node that originated the request (e.g. N19).

The NED function process ends when: (1) the depth limit defined previously is achieved; or (2) all possible paths between the nodes are explored by the NED function. Note that it is also possible to configure different depth requests in the scenario. For example, consider N2 with depth equal to 2. Then, N2 will send all the gathered information from the nodes N1, N7 to the chosen node N3, and then, node N3 will repeat the process to the next chosen node N8, which reached the depth limit. Node N8 will back track the information gathered up to the node that originated the depth request (e.g. N2). In the case that different NED paths converge to the same node (e.g. N8), this node is considered to be a cooperation point and will back track all information gathered, e.g. from the paths with depths at 2 and 3, to the nodes that originated the requests (N2 and N19). After this process, both nodes (N2 and N19) will synchronize and update the partial views with their directed connected nodes, containing all nodes and its gathered neighborhood hop-by-hop from the NED path

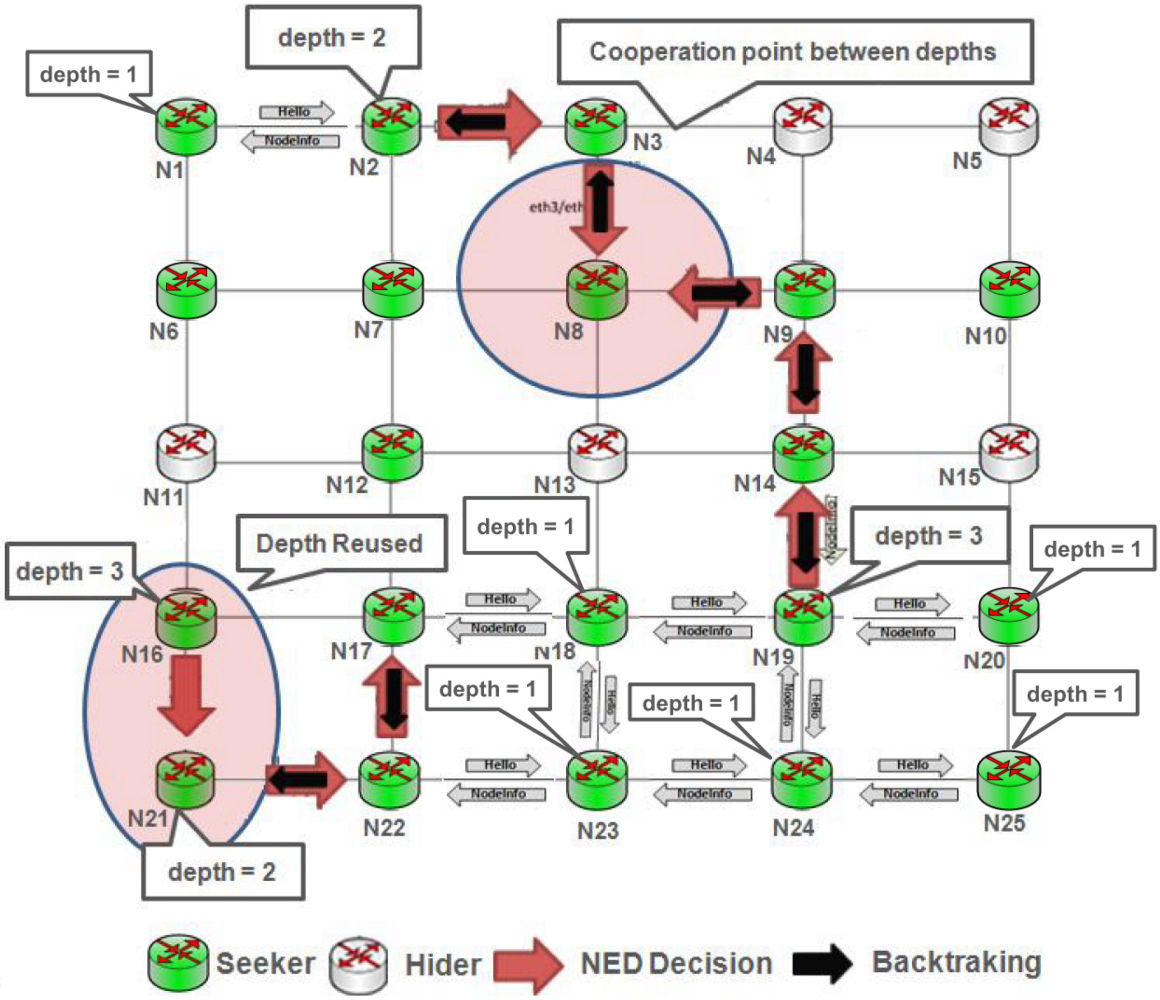

Figure 3.11: Example of Hide and Seek through the NED on static wired grid scenario

direction (e.g N3, N8, N9, N14). After that, all directed nodes already synchronized are able to exchange and cooperate according to a new requested depth. In addition, in any given time, there may be multiple cooperation points in the network, increasing the efficiency of information cooperation process. Moreover, the NED function can also reuse the knowledge from a node (e.g. N21) without making any further demands. For instance, N16 requests the knowledge with depth equal to 3, and N21 already has this information; then, N21 can reuse the information previously gathered in order to reply to N16. To maintain the information always updated, the partial views of each node consider the most recent information exchanged as well.

In order to complement the basic operating principles of the NED in wireless scenarios, in Figure 3.12, $N1$ is considered a dissemination device, where it is started the dissemination process to the neighbor devices $n2, n3...n11$. All devices directly connected to $N1$ send and receive the management flow informations. We consider management information as all type of data gathered from devices that compose the network, such as: identifiers, network topology, devices' resources status, bandwidth available, delays, network density, network addresses and domains. Then, the direction chosen by the NED function is based on the maximum rank of the Equation 3.5 to continue the dissemination process. So, the best direction chosen to

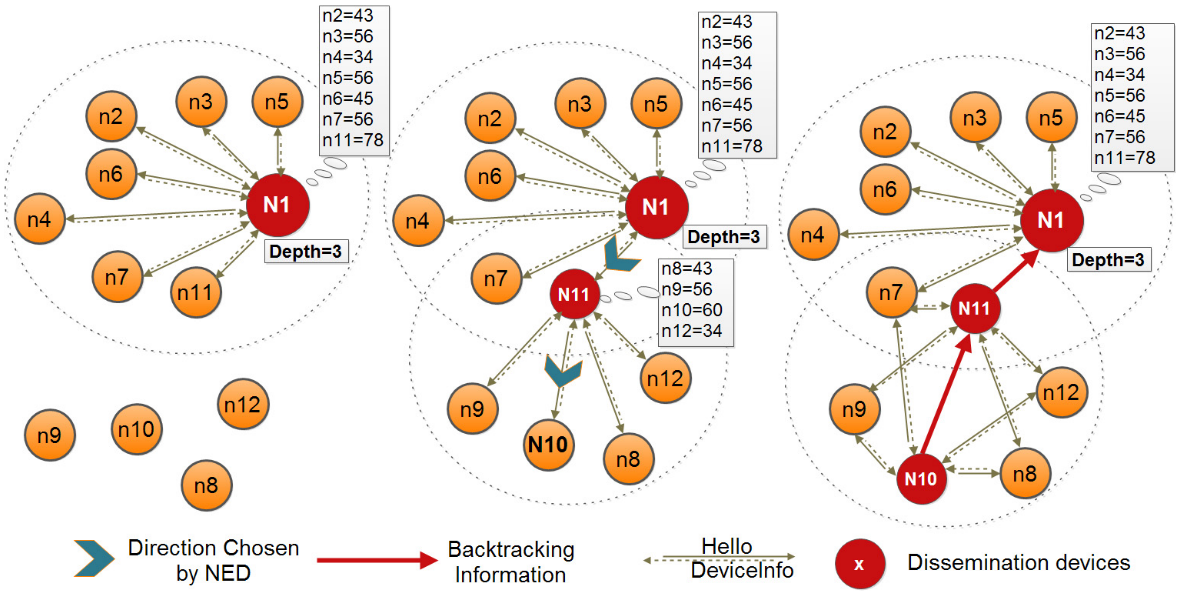

Figure 3.12: Basic operating principles of NED on wireless scenarios

continue the dissemination is the device $N11$, and information about $N1$ and its neighbors is disseminated to $N11$. Note that, the density of the $N11$ and $n7$ are the same, but $N11$ probably has better bandwidth available and local/network resources than $N7$, thus, the NED opted to choose $N11$ as the relay node.

The device $N11$ repeats the process and chooses a new direction (e.g $N10$). After this process, the device $N11$ also disseminates the management information to the next hop chosen, and finally all the information from $N1$ is disseminated through the neighbor devices. All information collected from $N10$ to $N11$ is back tracked to all devices that belong to the NED paths (e.g. $N11$) up to the device that originated the request (e.g. $N1$).

The NED function process ends in the same way as in wired scenarios. The NED can also reuse the depth from a device without making any further demands.

On the other hand, devices and links may fail for several reasons, such as devices hardware overflow (e.g. CPU and memory exhausted), nodes leave coverage area of each other, bandwidth bottlenecks or highly congested links. In Figure 3.13, it is presented an example of the dynamics when (a) node and (b) link failure are detected. For example, the device N1 (Figure 3.13 (a)) fails losing all the connectivity between the neighbors N2 and N4, whereas if just one link fails (Figure 3.13 (b)), N1 is still operational and can advise other devices (N2, N4) about its link failure to the device N8.

In Figure 3.13 (a), the device N1 fails; the devices N4 and N2 detect that N1 stops advertising its presence for a while. Both N2 and N4 check their local repository information gathered from the discovery process, if there is information larger than 1-hop. In case of N4, it has only information about the N1, which removes it from its local repository. On the other hand, the device N2 has information gathered from other devices in the network (e.g., N1 at 1-hop up to N9 at 5-hops). Then, the device N2 sends a recover/update message until the last hop device (e.g. N9 at 5-hops). The path that the recover/update message follows takes into account the chosen path already made by the discovery through NED function, avoiding extra messages propagation in the process. In fact, only the device N2 has next hop devices to relay the recover/update message. Each node that relays the recover/update message (e.g. N5, N6, N7, N8) updates its local repositories (about failed) N1 until reaching the last hop device (e.g.

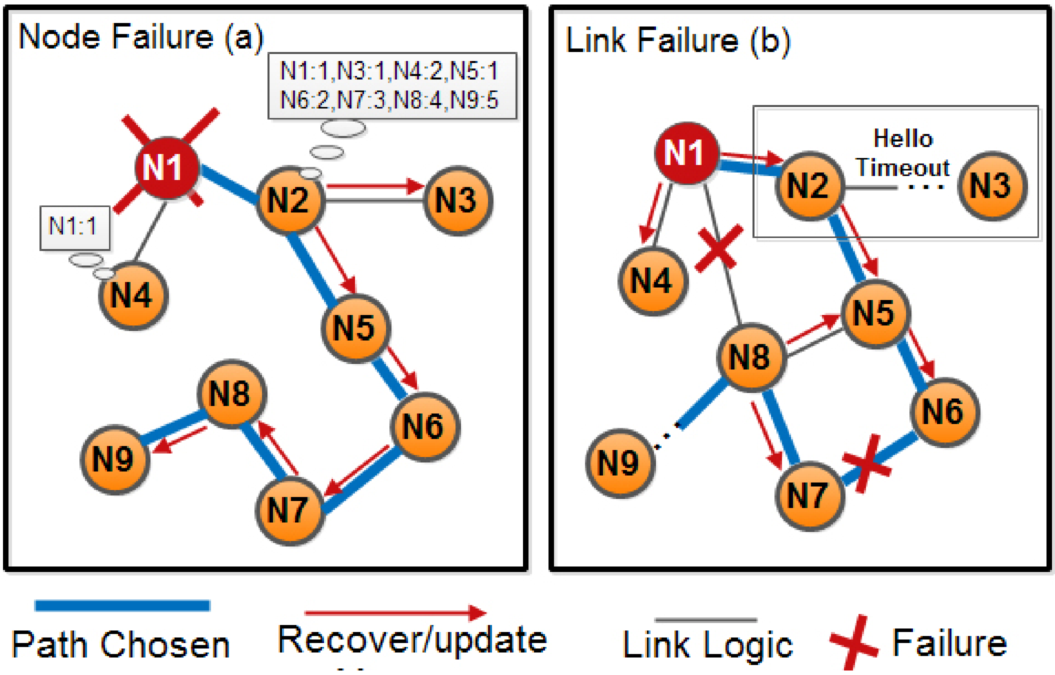

Figure 3.13: Example of Fault-Management Functionality for Nodes (a) and Links (b) failure

N9). Otherwise, if N1 recovers from failure, N2 advises the same way the existence of the N1 in the network. The failure can occur at any time and at any node, so the neighbors directly connected to the node take all appropriate actions whenever possible. Moreover, if a failure happens on the path already chosen by the NED function, the node that detected the fault (e.g. N6 and N7) has the possibility to call the discovery process again with the goal to find more alternative ways in which management information can be exchanged and recovered in case of failures.

In case of link failures Figure 3.13 (b), let's consider the case where the node has more than 1 link logic between the neighbors, e.g., N1 is directly connected to N2, N4 and N8. Then, N2 and N8 detect their directly link fails, thus, both N1 and N8 propagate a recover/update message to remaining links connected until reaching the last hop devices, similarly in Figure 3.13 (a). The devices N3 and N9 are moving out of the coverage area of the devices N2 and N8; thus, N2 and N8 detect also a link failure according to the HELLO timeout limit configured at bootstrapping device stage (e.g. limit detection time of 5% more than the HELLO time interval). Thus, the devices are warned without compromising the control performance of the network in comparison to the uncontrolled flooding techniques. Each recover/update message received by the devices updates the local repositories to the new information received, and makes the necessary updates automatically according to the acquired knowledge of the network. This ensures that the management information is always updated whenever it occurs any critical failure in the network.

3.3.1 Implemented modules and Interactions

The HISK2D with NED process is written in C/C++ programing language and consists of modules with distinct functions as can be observed in the Figure 3.14. Two main blocks are proposed: events handler which addresses the main events occurring in the system, and repository structures where they are stored in the node.

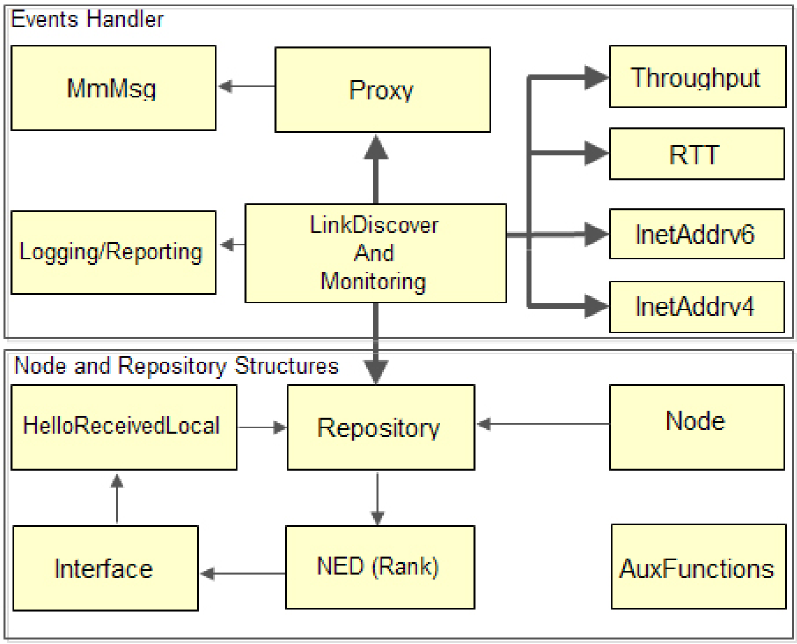

Figure 3.14: Functionalities of the discovery and exchange of management information

3.3.1.1 Functions

LinkDiscover is the main function of the discovery process which contains all data structures responsible to trigger the discovery and exchange of management information functionalities. Its execution is launched in each existing device interfaces, including both wired/eth0 or wireless/wlan0. In case of depth larger than 1 hop, the module *LinkDiscover* is responsible to forward the Hello packets through the interface which returns the value calculated by the NED. These packets are periodically sent by the device in order to discover other devices in the network (see Subsection 3.3.1.2 for more details on the Hello packet structure). Additionally, besides *LinkDiscover* function, HISK2D with NED integrates the monitoring function in the same process. This function is responsible for monitoring, detecting and reacting according to events or anomalies in the system. For example, the detection time limit of the Hello messages can be configured at the bootstrapping process. The fault-management process can advise other devices through recover/update messages (see Subsection 3.3.1.2) for more details on the Recover/Update packet structure).

The module *NED (Rank)* stores the ranking of the known devices calculated by the NED function to determine which is the best candidate neighbor to forward the information.

The module *Repository* is local to each device and it is composed by the partial view of the device. The information on each partial view entry is received from the device within reach and can also be updated according to topology changes. Therefore, the partial view is dynamic and its size varies according to the new received information. In addition, if the information is outdated (i.e. the node did not receive any contact in a specific amount of time), it assumes that there are no nodes in range.

The *HelloReceivedLocal* module records the devices identifiers from which Hello packets, which are defined in Subsection 3.3.1.2, have already been received, in order to reply to them with the appropriate information of the device. The module *AuxFunctions* is defined as a set of helper methods to ease information acquisition tasks. For instance, the methods to obtain the MAC address from the name of the interface, the percentage of free CPU and Memory

RAM, a list of all network interfaces of a specific device, a list of IPv4 and IPv6 addresses of an interface, etc. The module *Device* comprises the data structure of the devices, including the unique identifier, network status, devices resources and the rank calculated by the NED. The module *MmMsg* contains the definition of the message structures that are exchanged as well.

When a control packet for discovery and information exchange is received, the *Proxy* module will process it and send a response message back to the device that originated the request. In the case of depth larger than 1, the *Proxy* module is responsible to check if the message is at the last hop and, if so, it responds with the device information that rolls back to the originated device. The module *Logging and Reporting* will return local devices and network feedbacks in case of failures or anomalies e.g., invalid socket, full buffers, out of memory access, interfaces down, crashes on link/device and empty repositories. Modules as *Throughput* and *RTT* collect information about the status of packets and the links in the network, which will be then exchanged between the devices, helping to build the NED (Rank). *InetAddrv4* and *InetAddrv6* modules are responsible to convert automatically all signaling processes, which guarantee the consistence between both technologies (IPv4 and IPv6).

3.3.1.2 Packet Structures

This Section presents the structures of the exchanged packets.

The Hello packet, presented in Figure 3.15-a), is sent by each device in order to receive the contact of the other devices in the network. In addition, the time interval to send a Hello packet is adaptive according to the presented Equation 3.1 from Section 3.2.

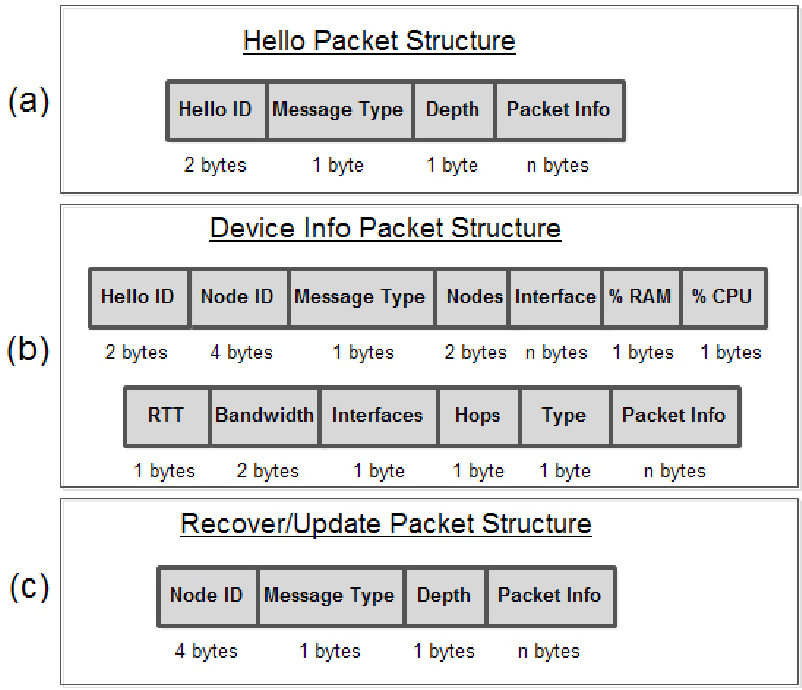

Figure 3.15: Packets Structure: hello packets, device informations and recover/update

The *DeviceInfo* packet is shown in figure 3.15-b) and it contains the information for the NED decision, which is sent by each device after being contacted, i.e. upon receiving a Hello.

In this packet the local resources information of the device is sent, as well as the controlled list of NED decisions, which synchronize the unexplored paths with the Hello Packet Info field. The *Devices* contain the number of known neighbors in order to calculate the value of $\mathcal{N}_i$ from Equation (3.4). The *Interface* field contains the interface, the device ID and IP from where the message was sent. The *% RAM* contains the amount of memory RAM and free *% CPU* available as well as the value of *RTT*, in order to measure the value of local and network resources reference value $\mathcal{R}_i$ from Equation (3.3). *Bandwidth* field contains information about the link capacity of the devices which serves as input value for variable $\mathcal{BA}_i$ from Equation (3.2). *Interfaces* indicates the number of local interfaces of the device. The *Hops* indicates from how many hops the original request came. The *Type* refers to the message subtype of a received DeviceInfo (1: initial exchange of messages by the directly connected devices; 2: reserved for exchange of information with depth larger than 1 hop; 3: updated the directed connected devices with the information gathered after finishing the process of exchange of information with depth larger than 1 hop). This ensures that the entire signaling process is performed through Hello and DeviceInfo messages, which are handled and forwarded internally by the *Proxy* module between the direct connected devices, or relay devices chosen when the depth is larger than 1 hop. Finally, the *Packet Info* is similar to the Hello messages, i.e. it contains the information of the devices through which the message will be forwarded back (e.g. backtracking) in case of depth larger than 1 hop.

The Recover/Update packet is shown in figure 3.15-c) and it contains the information for recover or update the devices when failures are detected. The *Node ID* contains the ID of device that originates the failure. The *Type* refers to the message subtype of a received Recover/Update (1: propagate error recover; 2: update local repository; 3 device is operational again). *Depth* is the number of hops to propagate the Recover/Update messages between devices. The *Packet Info* contains the information of the devices through which the recover/update message will be forwarded in case of depth larger than 1 hop.

3.3.2 Experimentation and Simulation Results

This Section presents the experimentation results of the HISK2D with NED and simulation results of the NED function at different depths, and in very large simulated scenarios.

3.3.2.1 Experimentation Results

The experimental set up is the same as in the previous Section 3.2.1. In this experimentation we consider a static grid topology of 9x9 with 81 virtual nodes. The NED weights are considered to be 33% for each one: link capacity (w1=1), local node and network resources (w2=1) and number of interfaces (w3=1). Where '1' means that the weight of the function is enabled. This analysis is performed on each machine varying the depth of the NED dissemination from 1 to 3 hops, and considering all nodes as dissemination points. We compare the NED dissemination with the dissemination mechanism of OSPF version 2. The OSPF is the mostly used solution to nodes discovery in wired scenarios and it was the one that showed the best performance when compared to CDP and OLSR (Section 3.2.1). The depth in OSPF is set using network areas: each node will have its interfaces configured to belong to a certain area depending on the desired depth, i.e., higher values of depth imply larger OSPF network areas. Regarding to the other protocols presented in the Section 3.2.1, CDP only sends information to directly connected nodes, and OLSR is designed to have better multi-hop performance in wireless scenarios, thus, they are not considered in this experimentation.

Additionally, we consider the management information as: number of nodes, ID/MAC, IP, RTT, number of interfaces, CPU and memory available, bandwidth and number of hops.

The evaluation metrics considered are the following: (i) convergence time, which quantifies the time to discover the nodes according to the configured depth (i.e. lower is better); (ii) amount of bytes exchanged, which quantifies the amount of control packets sent by the protocols in the network (i.e. lower is better); (iii) number of control packets, which measures the total amount of control packets sent by the protocols (i.e. lower is better); and (iv) average number of disseminated nodes, which quantifies the number of nodes that received the management information according to the configured depth (i.e. higher is better).

The results are obtained considering the applicability of the HISK2D with NED function to discovery and disseminate information at different depths: 1, 2 or 3 hops. The values presented in the graphs below are an average of 5 repetitions and 95% of confidence interval. To perform the packets capture, TCPdump is the software used in each node interface, with an observation time of 10 minutes. The links capacity is considered to be 1 Mbps full duplex.

This analysis is performed on each machine varying the depth of the discovery process from 1 to 3 hops. In HISK2D with NED, this is directly configured in the NED function, which will determine the best interface through which the discovery should proceed.

Figure 3.16 shows that HISK2D with NED requires less management control packets than OSPF, regardless of the chosen depth. This is mainly justified by the different roles in HISK2D process, which means that a node can be a seeker or a hider at a given time. Thus, when a node is in hider mode, it does not signal packets, but waits to be contacted for a seeker node or starts itself the discovery process after a given time, which is configured at the bootstrapping process of the node. This behavior reflects in the lower amount of control packets exchanged (Figure 3.16). More detail on the Hide and Seek process can be seen in the Section 3.2.

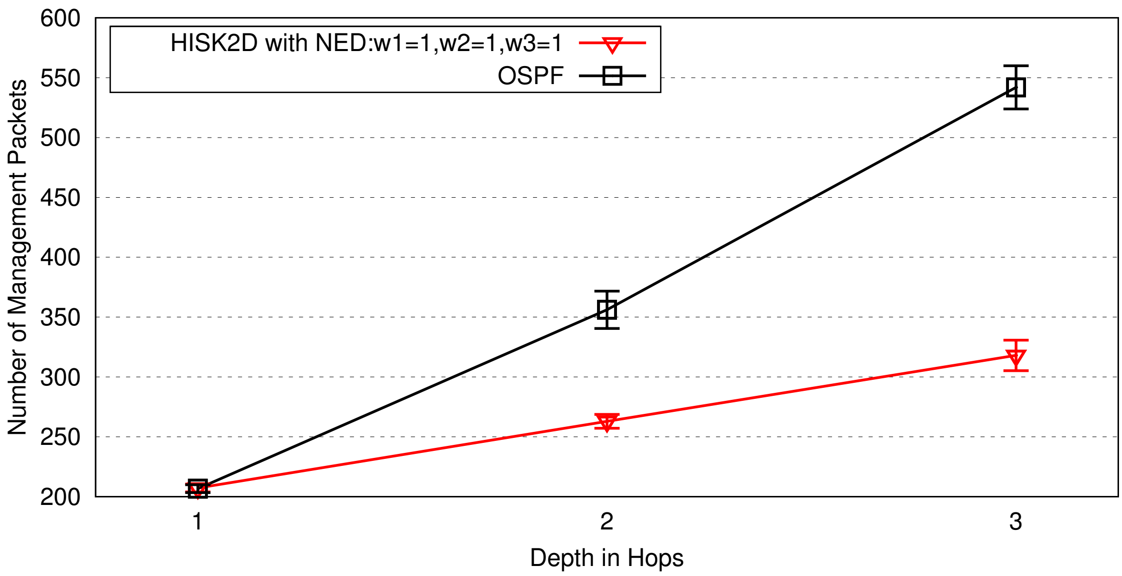

Figure 3.16: Grid 9x9: Number of Control Packets Exchanged

Figure 3.17 analyzes the amount of bytes exchanged to discover/disseminate management information according to the configured depth. The HISK2D with NED has less bytes exchanged to discover and disseminate information than the OSPF. This is due to the fact that the NED function is enabled when the depth is larger than 1 hop, causing the dissemination process to follow the chosen paths. When the flow reaches the predefined depth, nodes crossed by the flow are unable to send update messages until the backtracking process has been finished. This does not necessarily mean that HISK2D with NED exchanges less information than OSPF, but rather that no redundant data is used, i.e., only the strictly required one is forwarded between nodes. The periodic updates only include the parameters that are changed in the meantime. Thus, if a new node enters or leaves the network, or if the fault-management process detects an abnormality in a node or in a link, the node that detects it only send periodic updates containing the new information in order to refresh the local repository of the nodes that belong to the NED path. This behavior jointly with the explanation of Figure 3.16 bring to the HISK2D with NED a lower signaling in the network communication.

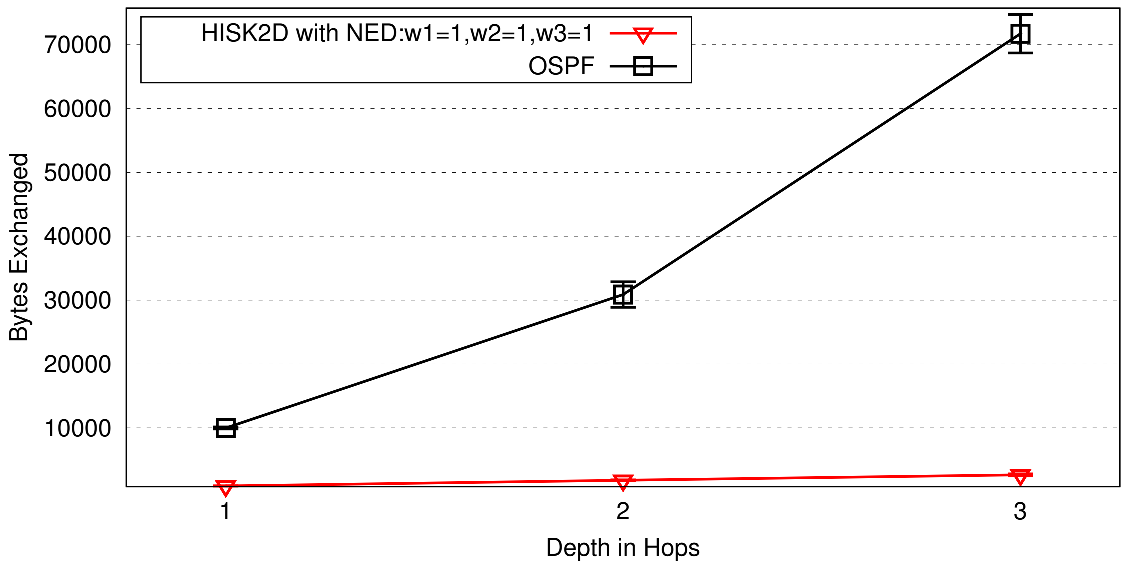

Figure 3.17: Grid 9x9: Bytes Exchanged to disseminate management information

Figure 3.18 presents the convergence time. Setting any of the presented depth values, HISK2D is, at least, 60% faster than OSPF. This behavior is explained due to the fact that OSPF uses Dijkstra's algorithm to calculate the paths, which is complex and consumes significant time. In case of HISK2D with NED, the cooperation between nodes that belong to the NED path updates its paths with the directed connected nodes, thus, reducing the time to find a required path.

Figure 3.19 shows that the average number of disseminated nodes by each node is similar using both protocols. Therefore, HISK2D with NED can achieve the same results of OSPF using less network resources regardless of the desired depth.

3.3.2.2 Simulation Results

The behavior of the NED dissemination is also analyzed in MATLAB [MAT11] simulations, in dense static wireless networks, from 100 up to 500 devices, placed with realistic models. We also performed an evaluation using random placement models in wireless dense scenarios which is detailed in the Annex B.

The wireless transmission range of the devices is defined with 100m. The realistic model adopted for devices placement in this simulation is based on the topology type NPART/Berlin (from NPART [BM09]), which simulates the placement of the wireless devices in the city of Berlin, adjusting the simulation area (500m x 500m).

The uncontrolled flooding and Gossip [Mat05] approaches are used as the baselines, since they are the most adopted approaches for comparison purposes in the literature. The un-

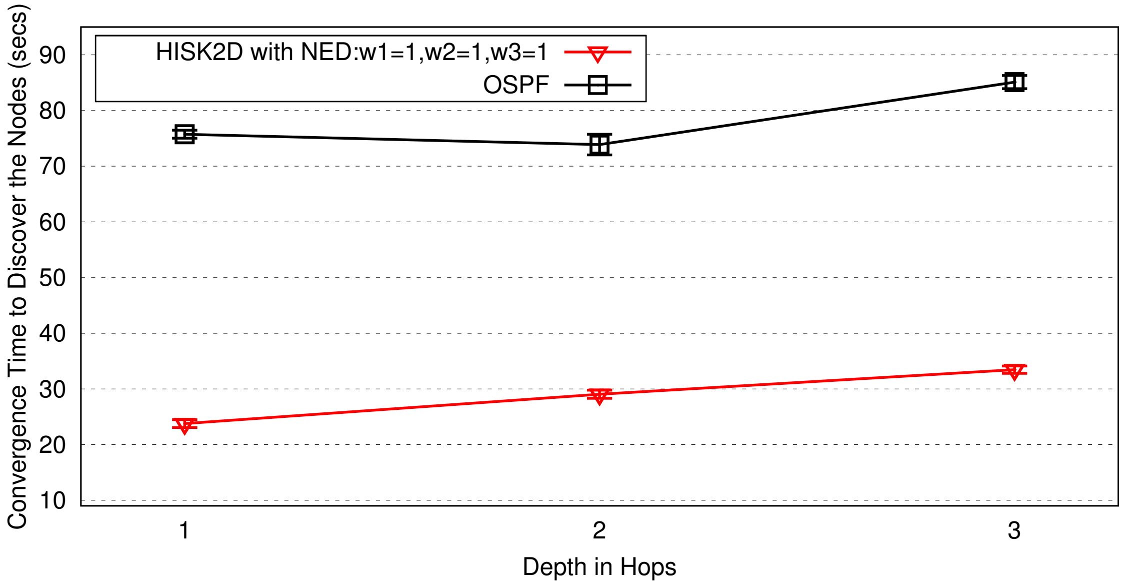

Figure 3.18: Grid 9x9: Convergence Time in Seconds to Discover the Nodes

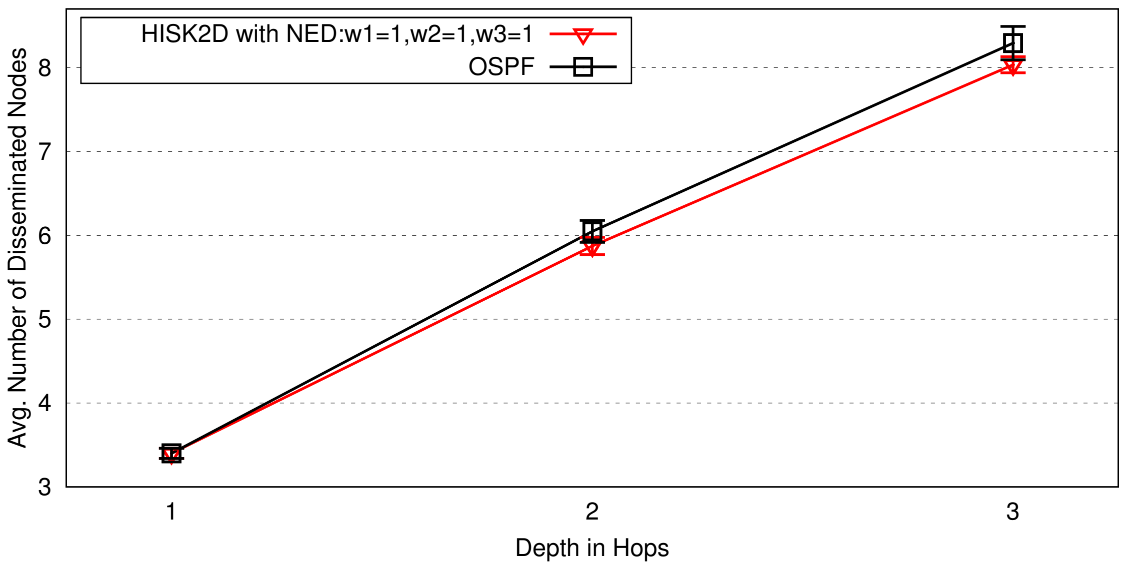

Figure 3.19: Grid 9x9: Average number of disseminated nodes

controlled flooding implementation is based on [Mat05], but without flooding control. The evaluation is performed in an area of 500m x 500m, changing the number of devices and the number of dissemination devices, where it is ensured that no device in the network is isolated.

The number of dissemination devices, which is the number of devices that start the dissemination process, is defined as 10% of the total number of devices in the scenario (e.g. 300 devices has 30 dissemination points), uniformly distributed through these devices. Notice that the number of dissemination devices is perform only for NED and Gossip, since in the uncontrolled flooding all the devices act as dissemination devices.

In the Gossip, the maximum number of neighbors to perform the flooding dissemination is 25% of the dissemination devices, in order to avoid unnecessary and uncontrolled flooding (e.g. for 20 dissemination devices the Gossip probabilistically chooses only 5 neighbors to flood the management information).

The NED results are obtained from different experimental-driven combinations (111 | 100 | 010 | 001) of the weights $w1$, $w2$ and $w3$ (see Equation 3.5), and for a dissemination's depth much higher than the number of wireless devices. The results are obtained from 10 repetitions with different seeds and a confidence interval of 95%.

For the evaluation of the three analyzed approaches, three metrics are proposed, dissemination cost, recursive cycles, and disseminated devices. The dissemination cost is defined as $Dc = 1 + \sum nsize$, where 1 is the broadcast discovery messages and $nsize$ is the number of neighbor devices in which the management information is disseminated. The redundancy messages are also considered, but they depend on each algorithm (e.g. uncontrolled flooding has a higher probability to have redundancy messages, since all devices perform flooding). In terms of the analysis, a lower dissemination cost is better. Recursive cycles measure the impact in terms of local device resources consumption to execute the algorithm. When the algorithm recursively sends broadcast messages to the neighbor devices, it is counted as 1 cycle. The lower number of recursive cycles means less device resources consumption (e.g. CPU and Memory). Disseminated devices measures the number of devices that receive the dissemination messages. The higher the number of disseminated devices the better.

NED reduces the dissemination cost when compared with Gossip and uncontrolled flooding in realistic (Figure 3.20) scenarios, especially with the growth of the number of devices in the network (e.g. up to 200). The uncontrolled flooding and Gossip have a higher cost in both scenarios to disseminate the same management information in the network.

The behavior of NED is explained by the cooperation between devices, according to a dynamic adjustment of the depths functionality, i.e., devices do not necessary need to reach the whole network until the maximum depth value. For example, when NED exhausts the possible paths to continue the dissemination, only the device that belongs to the NED path will disseminate the management information, contributing to an efficient dissemination of messages. On the other hand, Gossip performs controlled flooding, which introduces an extra cost proportional to the number of devices in the network. The Gossip behavior is explained by the probability function used to define how many neighbors should the information be flooded to. This probability is calculated according to the number of devices directly connected, where the maximum number of neighbors to perform the flooding is defined to be 25% of the dissemination devices. Uncontrolled flooding has the highest number of messages, since all devices flood the management information.

The weights combination of NED does not have a significant impact on the dissemination cost for both scenarios. In fact, for each repeated simulation, the bandwidth and local resources for each device are regenerated without any pattern, reducing the influence of NED weights ($w1$,$w2$, $w3$). The weight $w3$ should have more influence in the dissemination cost, but due to the nearly uniform placement of the devices in both scenarios, it balances the network density value (Equation 3.4) for all the device's neighbors.

The analysis of recursive cycle (Figure 3.21) is important to assess the impact of the three approaches in terms of resource consumption (e.g. memory usage and cpu processing). The lower is the number of recursive cycles, the lower is the consumption of the device resources.

The uncontrolled flooding has the highest number of cycles, since all devices flood the management information through the network. The NED reduces the recursive cycles when compared with Gossip, since in Gossip part of the devices flood the network with management information, while NED chooses paths that prioritize bandwidth available and the devices' density, spreading more information with less depth between the dissemination devices. The

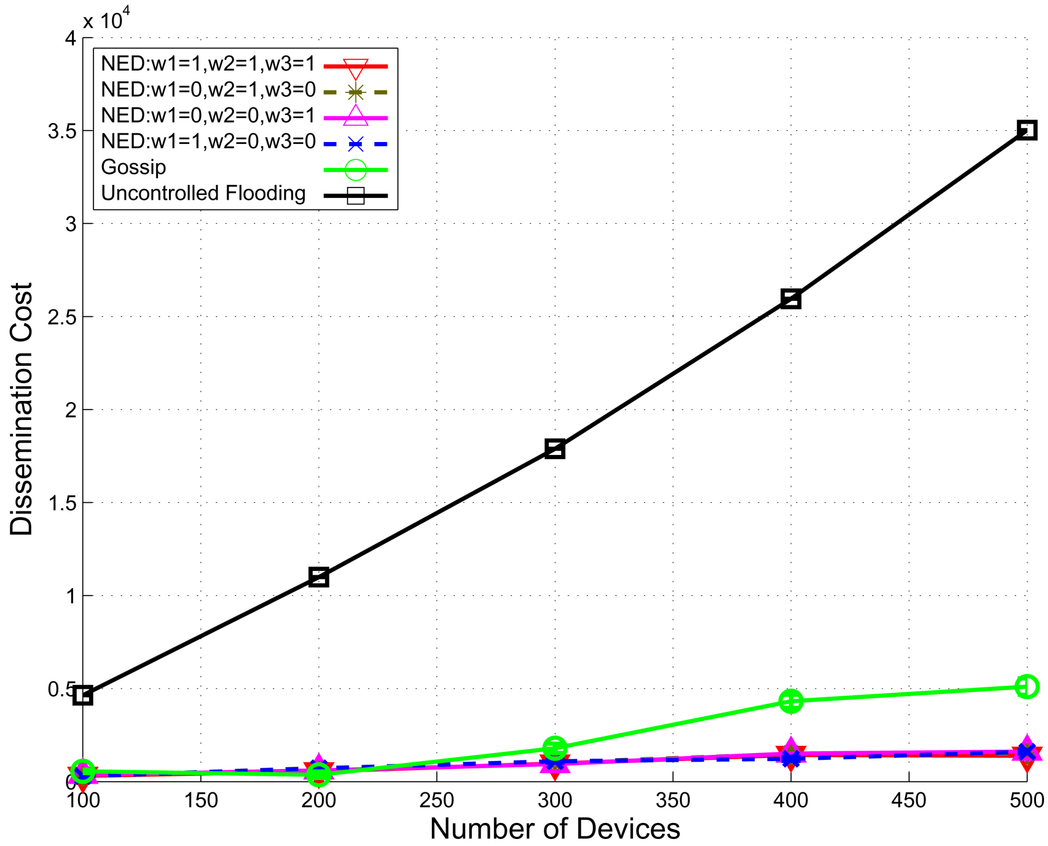

Figure 3.20: Dissemination cost in realistic scenarios

increasing number of devices enlarges the difference of the number of recursive cycles of the three approaches, since it is almost constant in NED due to the dynamic adaptation of the depth, while in Gossip and uncontrolled flooding it grows.

In Figure 3.22 only uncontrolled flooding reached the total number of disseminated devices. The Gossip depends on the amount of neighbor's devices that perform the dissemination of the management information (25% of dissemination devices), whereas NED depends on the dissemination devices to reach the total number of devices in the network. In this case, both 10% of dissemination devices and 100m of wireless range are not sufficient to reach all devices in the realistic scenario.

A trade-off between the number of dissemination devices and the disseminated devices is required, which should be achieved considering the wireless range of the devices and the considered area. Table 3.3 extends the analysis of the three evaluated metrics for the three approaches, evaluating the impact of the number of dissemination devices (e.g. 20% and 80%), and the wireless range (e.g. 50m, 100m and 150m) for 500 devices in a realistic scenario. We consider the NED weights ($w1 = 1, w2 = 1, w3 = 1$) in this evaluation.

In Gossip the increase of dissemination devices, and consequently, the increase of the maximum number of neighbors to perform the flooding dissemination strongly increases the dissemination cost and recursive cycles, especially for higher wireless ranges. In NED it reduces the dissemination cost and recursive cycles for higher wireless ranges.

From Table 3.3 we observe that NED with more dissemination devices decreases the dissemination cost and recursive cycles for higher wireless ranges (e.g. 100 and 150). For lower

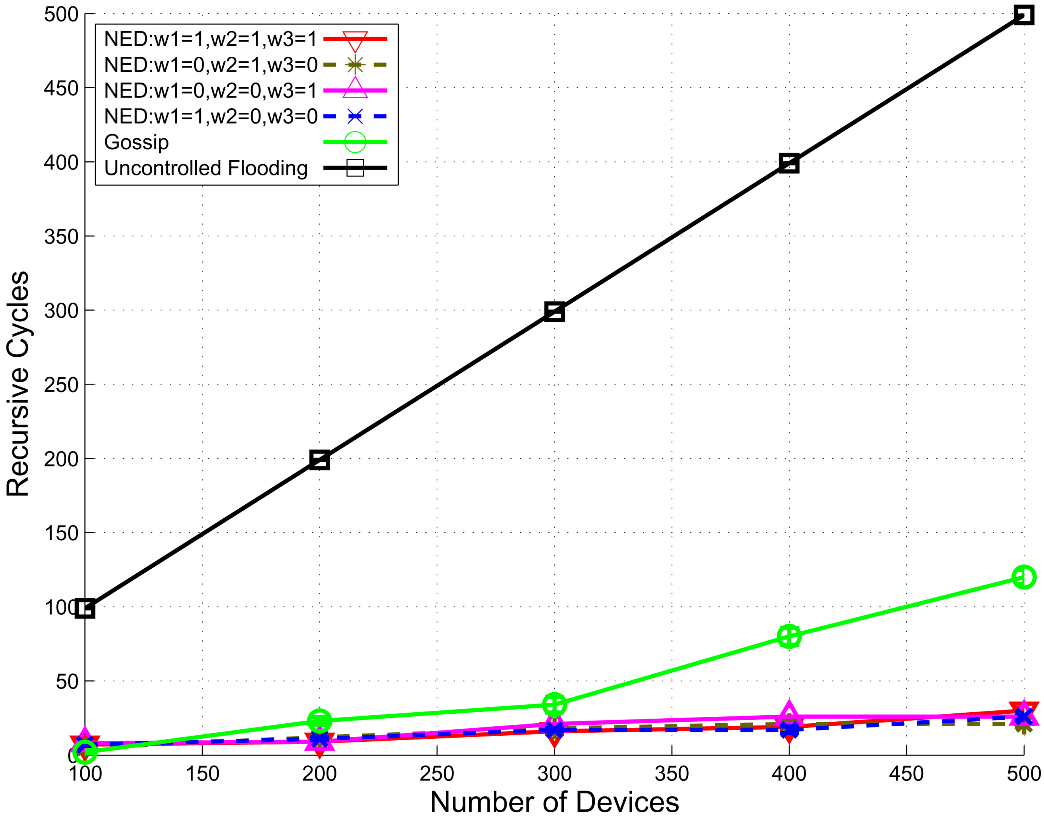

Figure 3.21: Recursive cycles in realistic scenarios

wireless ranges (e.g. 50), less dissemination devices decreases both dissemination cost and recursive cycles, but it does not disseminate the management information through all the devices of the network.

3.4 Network (Re) Organization

This Section is mainly supported by the Papers V and VI presented in the Table 1.1 from Chapter 1, and Paper VI is presented in the Annex C.

Autonomic and distributed management requires the cooperation between the network nodes and the periodical exchange of network information between them. Thus, the network nodes need to be associated and grouped through a stringent set of rules, to minimize the amount of control messages flowing in the network. In order to overcome this requirement, a mechanism for association and connection between wireless nodes in communities through social-based metrics is proposed. To infer the technical behavior of users and predict their interaction in the communications, the emerging concept of mobile social networking [RO08, Lug08] can be re-though for network management support. In mobile networking, users access virtual communities and share their social interests between themselves. Recently studies [NDXT11, PP09] have shown how the social communities are formed and grouped by interests providing helpful information in order to develop social-aware strategies for social networks problems. These concepts may inspire the understanding of, for example, how wireless communities can be structured so that the users can share their personal interests. By the definition of new social metrics, the nodes are self-organized into quality communities

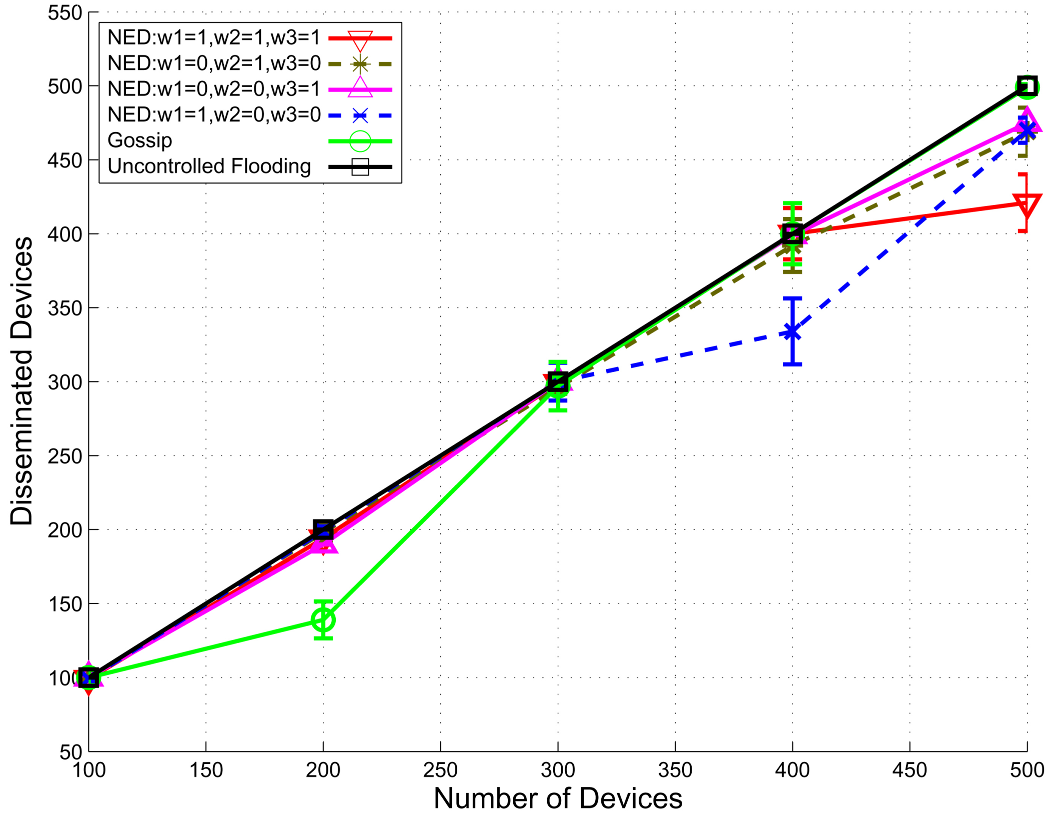

Figure 3.22: Disseminated devices in realistic scenarios

which improve their message exchange for the support of management decentralization in wireless networks. These social metrics measure and predict the Friendship of Neighborhood Nodes, Associated Nodes and Community Nodes, so that the connection between the nodes is performed through this set of metrics to optimize their stability and information dissemination efficiency.

The association process requires also the bootstrapping and discovery process, as aforementioned at previous Section 3.2. From a single node viewpoint, the bootstrap will set up the initial information, e.g. nodes identifiers, initial management policies strategies, assignment of unique identifier, MAC address, BSSID, hardware capabilities. After the bootstrapping stage, the node is ready to start cooperating in the discovery phase in order to identify the surrounding neighbors. We consider surrounding nodes the ones that can communicate with each other in the transmission range. The discovery protocol builds and periodically updates the information about neighbors, which is performed through an extension of the IEEE 802.11 MAC signaling process (e.g Beacons, Probes, Associations/Reassociation and Disassociation) and a set of new IEs [IEE07]. This information is used by the nodes to decide the best neighbor or community for association, according to the proposed social-based metrics (described in Section 3.4.3 and presented with greater detail in the Annex C). The discovery process is based on continuous search for new nodes, or on the update of the information of known nodes. The signaling process considers direct and indirect contact (e.g. transmission and reception of Beacons). The gathered information is obtained from the information received through the IE Beacons, and is stored in the Partial View Table (see Figure 3.23 with an example of the partial view), which now is extended with the social metric information. The

Table 3.3: Evaluation for realistic scenarios with 500 devices: 20% and 80% of dissemination devices; and 50m, 100m and 150m of wireless range

20% Dissemination Devices

NED	***Dissemination Cost***	***Recursive Cycles***	***Disseminated Devices***
50m	1294 ± 66.9619	77 ± 4.9660	450 ± 13.1572
100m	1905 ± 123.1851	29 ± 1.4001	500 ± 0
150m	1844 ± 74.1856	17 ± 0.6940	500 ± 0
Gossip	***Dissemination Cost***	***Recursive Cycles***	***Disseminated Devices***
50m	8495 ± 101.3967	476 ± 2.9209	500 ± 0
100m	11146 ± 376.1676	203 ± 6.5126	500 ± 0
150m	11990 ± 550.2752	105 ± 6.5126	500 ± 0

80% Dissemination Devices

NED	***Dissemination Cost***	***Recursive Cycles***	***Disseminated Devices***
50m	1682 ± 24.0675	98 ± 1.2180	500 ± 0.2594
100m	1885 ± 63.0196	28 ± 0.8139	500 ± 0
150m	1500 ± 93.3205	16 ± 0.4021	500 ± 0
Gossip	***Dissemination Cost***	***Recursive Cycles***	***Disseminated Devices***
50m	9823 ± 0	499 ± 0	500 ± 0
100m	35015 ± 0	499 ± 0	500 ± 0
150m	68549 ± 225.3114	489 ± 1.0637	500 ± 0

Known Nodes Table (see Figure 3.24 with an example of the nodes information) contains information about nodes found through other nodes in a specified range. The information on each partial view entry is received from the node within reach and can also be updated according to topology changes. Therefore, the partial view is dynamic and its size varies according to the new received information. In addition, if the information is outdated (i.e. the node did not received any contact in a specific amount of time), it assumes that there are no nodes in range.

NeighborID	SocialMetric	Timestamp	PViewSize	RecBeacons
5	0.96	132118321	10	0.6
2	0.65	154564455	5	0.3
9	0.26	16456464	8	0.1
...	...	...	...	...

Figure 3.23: Example of Partial View Table filled with social-metric information

In the Partial View Table, *NeighborID* and *SocialMetric* fields identify the neighbor node related to that Partial View entry and its current social metric; the *Timestamp* field allows the local node to know when the information is outdated; the last two fields, *PViewSize* and *RecBeacons*, are used to build the social metrics.

Due to the limited operating range, the direct contact between the nodes provides only a very limited knowledge of the network. Thus, it is required to introduce the Known Nodes Table that is built through cooperation between nodes to spread the knowledge of its neighbors. This local table is also dynamic, i.e., its size grows as new information is recorded, but it contains only information about nodes found through other nodes in a specified range. The need for this table relates to the fact that one of the social metric estimates the size and quality of the community to which the node belongs (See Section 3.4.3). The table is filled according to the information present in the vectors *Known Nodes* encapsulated in the Community-Based IE according to Figure 3.27. This results in a more extended knowledge of the network, allowing the nodes to have more information on which to base their decisions.

NodeID	CommunityID	Code	$\overline{\mathcal{G}_i}$
1	723	2	0.6
2	412	1	0.2
3	723	0	0.8
...	...	...	...

Figure 3.24: Example of Known Nodes Table

In the Known Nodes Table, *NodeID* and *CommunityID* fields identify the node related to that entry and the community where it currently belongs; the field *Code* is used in the proposed algorithm of community size estimation in order to maintain the consistence of information and it varies from 0 to 3; the last field represents the value of the social metric *Associated Nodes Friendship* $\overline{\mathcal{G}_i}$ reported by each node (see Section 3.4.3).

3.4.1 Communities Composition

A community is a set of nodes associated with each other, i.e., a community ensures that there is always an association path between any two nodes. This concept is important from the point of view of decentralized management, since the routing of data between any two nodes is guaranteed in a community.

A single large community occurs with associations not only within the same community, but also among the nodes in different communities. In addition, the node that performs the association adopts the identifier of the new community and propagates it to the nodes associated with it and so on, until all elements of the community agree on the current identifier, avoiding inconsistencies of the community identifier (Figure 3.25).

The loss of connection can occur not only at the extremes of the community, but also within the same. In this case, the community splits into two smaller ones. A community will maintain its identifier and the other will have to adopt a new ID (Figure 3.26). To this end, the node that looses the connection will be responsible for generating a new identifier and spread it to the nodes associated with it and so on, until all nodes agree on the current community identifier.

The communities are created and optimized in a distributed way, since each node is associated to the one it considers to be the best one in the surrounding neighborhood. In parallel, the process of maintaining the community takes place, since associations are not ensured to be definitive. Each time a node is contacted by a node whose social metric is larger than the one of its current association, it needs to signal the node which holds the current connection, indicating that it will end the relationship. Thus, the established associations will

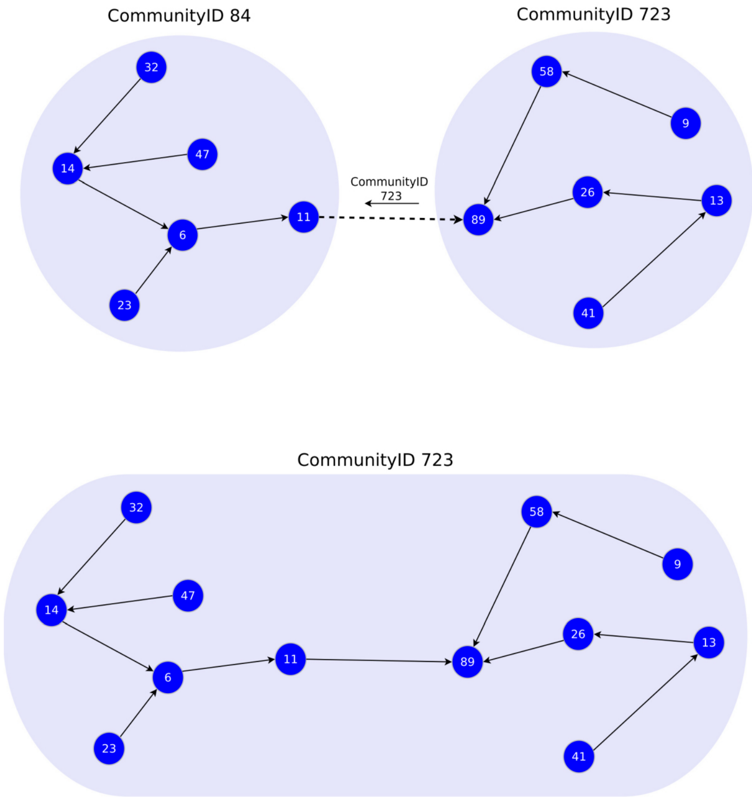

Figure 3.25: Communities Union

include only the best connections in accordance with the proposed social metrics. In order to avoid constant changes of associations by slight differences in the social metric, we defined a threshold up to which it is not triggered a new association process. The mobility of the nodes triggers changes in the calculated social metrics: this may result in the merge and split of the communities according to Figure 3.25 and 3.26.

Considering the IEEE 802.11 as the base standard and protocol, it is necessary to extend it to meet the requirements of the proposed mechanisms and social-based metrics. At the level of management frame Beacon and Association Response, the introduction of new IEs (see Figure 3.27) in the Frame Body field allows the nodes to cooperate in order to share informations between them (e.g. fill the local tables, Partial View and Known Nodes, with the knowledge of each node). This requires a new community-based IE as depicted in the Figure 3.27. The respective Beacon response is depicted in the Figure 3.28. With the introduction of decisions at local level, the association will be based on specific parameters in a deterministic way and not on the first node to send a Beacon.

Moreover, an adaptive Beacon interval is proposed and it is dynamically calculated using parameters of the social metrics (defined in Subsection 3.4.4). The aim is to ensure the efficient management of the network with a reduced message overhead.

As previously referred, the association between nodes is no longer made through first beacon approaches: the association is performed to the node with the highest social metric in the vicinity.

On the other hand, we assume that a community has a parent-child relation structure, where the node that initiated an association is the child of the other node (i.e. the parent).

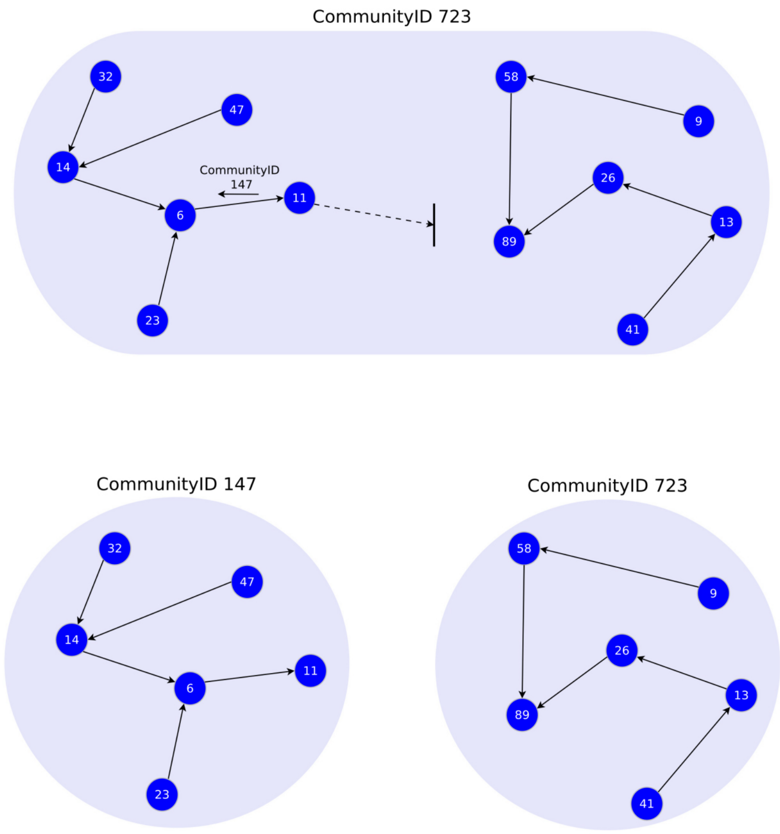

Figure 3.26: Communities Split

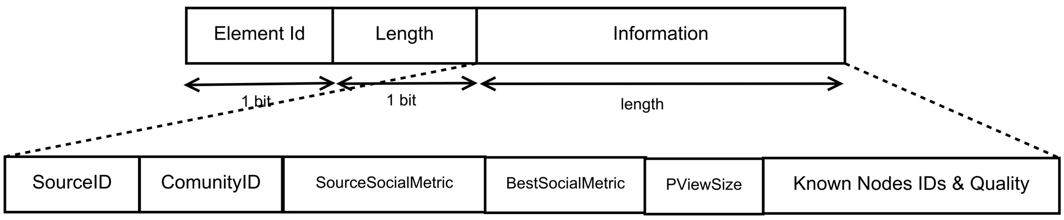

Figure 3.27: Community-Based Beacon Information Element

A distributed algorithm is developed in order to estimate the communities size. It uses a distributed backtracking process, where: (i) all nodes report to their parent node the number of nodes associated to them, (ii) the head node (i.e. the first node of the community and overall parent) processes all reports of its child nodes, estimates the size of the community and announces back the result to all its child nodes and (iii), subsequently, all nodes report to all their child nodes the size of the community announced by the head node.

3.4.2 Loop Avoidance Criteria

Loops can occur due to two different reasons: concurrent requests and reports of outdated information. The information achieved by the nodes might be non-synchronized, i.e, the knowledge is distributed across the network. This implies that a node may report something

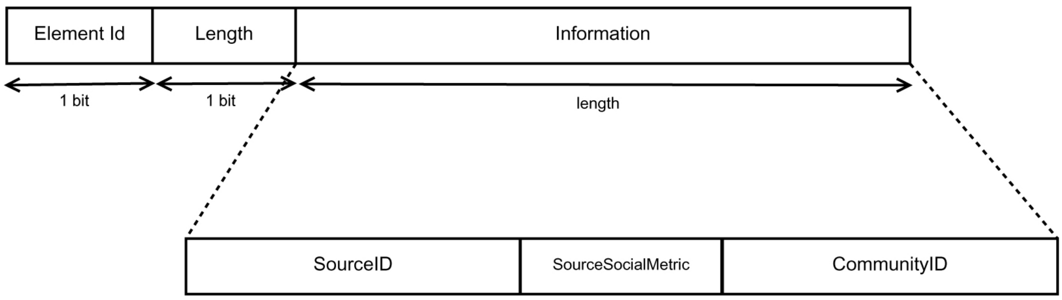

Figure 3.28: Community-Based Association Response Information Element

that is not true in that moment, because it has not already received the updated information. These time instants are sufficient to cause temporary or even permanent loops in the network due to misleading information.

In order to prevent loops, the main criteria implemented is the following:

- To prevent 1-hop loops, a node cannot accept association requests from the node that it is associated or is about to be, i.e., it has already sent an association request to this node (Figure 3.29).

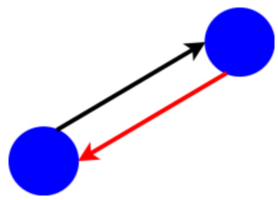

Figure 3.29: 1-hop loop avoidance

- In order to avoid loops due to concurrent association requests, a node that has requested an association and receives a request in the meantime, cannot reply to it positively (Figure 3.30).

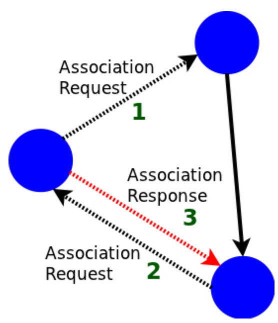

Figure 3.30: Loop avoidance in concurrent events

- For multi-hop loop prevention, two situations are considered:

  - when the node belongs to a community, it can only change its current association to another node within the same community if it gets closer to the parent node in terms of hops count (Figure 3.31). By this way, it is impossible for any node to change its association to a node that is behind it, i.e., in the same branch, which can lead to a forced disassociation from the current community and the consequent loop creation.

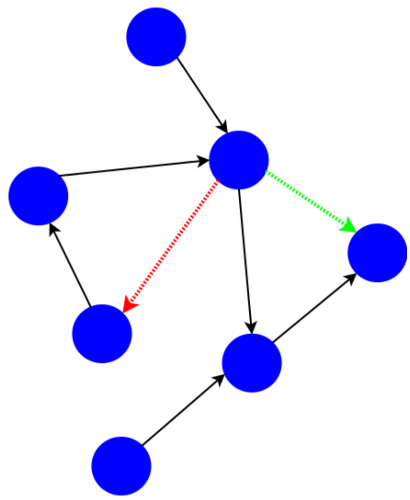

Figure 3.31: Multi-hop loop avoidance

  - when the node has lost several Beacons (i.e., if the watchdog timer set up reaches the end), it assumes that its association no longer exists and becomes the parent of a new community (the group of nodes associated to it). Due to the propagation time of the new parent ID, it cannot associate to the previous community of nodes because it is not guaranteed that it will not create a loop with a node that did not already received the new parent ID.

- Even with the previously referred criteria, there are still possible situations where it is impossible to prevent the loop creation. These situations are mainly related to non-synchronized information between nodes. Therefore, for multi-hop loop detection and removal, the criteria is that if a node receives a community ID generated by itself, coming from the node to whom it is associated, it proves the existence of a loop. Therefore, the node that detects this, will be responsible to finish its current associations, thus breaking the loop (Figure 3.32).

It also is important to refer that the community ID is composed by the MAC address of the node who generated it and a random number (Figure 3.33). If a node receives a community ID that it has generated, coming from the node to whom it is associated, it means that somewhere in the community exists a loop. Therefore, this node will break the loop by ending its association.

To reduce the probability of non-synchronized information between nodes within the same group, namely the Parent ID that can lead to loop creation as previously explained, a quick reverse mechanism has been implemented. The mechanism is activated whenever the parent ID changes, in order to minimize the time during which the nodes have incoherent information. It consists in Beacons with updated information that are forwarded by the nodes in the reverse

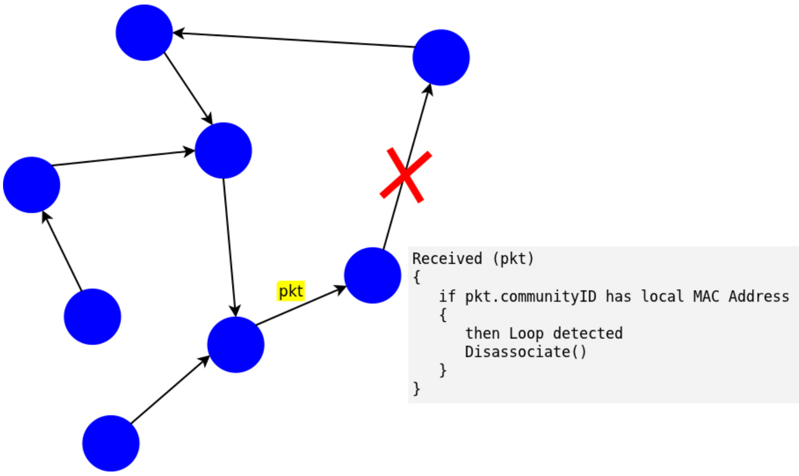

Figure 3.32: Multi-hop loop detection and removal

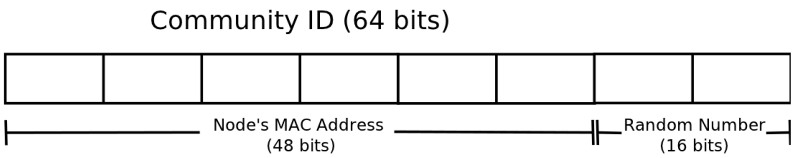

Figure 3.33: Community ID composition

direction of the association, i.e., from the parent node to the child nodes. Each node will extract the information from the packet received and will update its local knowledge. By this way, when this flow reaches the farthest node from the parent, all nodes of that branch are synchronized.

Finally, in the next Section 3.4.3, we will define the proposed social-based metrics.

3.4.3 Social-Based Metrics for Nodes Association

Our model assumes a distributed ad-hoc mobile wireless network consisting of nodes that interact with each other through direct or indirect contacts, forming a fully connected network in the coverage area, where nodes have at least one common channel between their neighbors.

We define $\mathscr{N}$ as the complete set of nodes in the scenario:

$$\mathscr{N} = \{n_i : i = 1, \ldots, N\} \tag{3.6}$$

where N represents the total number of nodes. In any given time instant, all nodes have a partial view of the network, i.e., the set of nodes in the neighborhood of a node i (partial view of node i) can be defined by:

$$\mathscr{P}_i = \{n_j : n_j \in \mathscr{N} \wedge j \neq i \wedge v(i,j) = 1\} \tag{3.7}$$

where $v(i,j)$ is a function that returns 1 if nodes i and j are neighbors and 0 otherwise. The subset of nodes associated with node i can be defined by:

$$\mathscr{A}_i = \{n_j : n_j \in \mathscr{N} \wedge j \neq i \wedge a(i,j) = 1\} \tag{3.8}$$

where $a(i,j)$ is a function that returns 1 if nodes i and j are associated and 0 otherwise. Note that, $\forall i, \mathscr{A}_i \subseteq \mathscr{P}_i \subseteq \mathscr{N}$.

In any given time instant, a node i belongs to a community of nodes. A community of nodes is composed by nodes that are directly associated or have a multi-hop path between them. The subset of nodes in the same community of node i can be defined by:

$$\mathscr{C}_i = \{n_j : n_j \in \mathscr{N} \wedge c(j) = c(i)\} \tag{3.9}$$

where $c(i)$ is a function that returns the identification of the community of node i. Note that, $\forall i, \mathscr{C}_i \subseteq \mathscr{N}$ and $\forall i, |\mathscr{A}_i| \leq |\mathscr{C}_i| \leq |\mathscr{N}|$, where $|.|$ represents the cardinality of a (sub)set.

Problem Formulation: *Let $n_i \in \mathscr{N}$ be a node that periodically sends Beacons to its surrounding neighbors ($\mathscr{P}_i$). In any given time instant this node is associated with the set of nodes $\mathscr{A}_i$ and belongs to community $\mathscr{C}_i$. However, this node shall be able to decide the best criteria of association taking into account the quality of its community, as well as the degree of friendship/quality of its surrounding neighbors and communities. The mobility of its surrounding neighbors, as well as the constant maintenance of these best association criteria, needs to be considered.*

We adopt the following assumptions to define the social metrics in the model:

- All nodes have a unique identifier;
- Nodes are assumed to be mobile;
- The standard criteria for association is the first node to receive a Beacon;
- Nodes need to be in range to be able to communicate;
- All nodes are equipped with one radio transceiver (transmitter and receiver) capable of either transmitting or receiving at any given time, but not simultaneously;
- The antenna model is omni-directional with maximum cover area using the *Friis* Model;
- The MAC level is assumed to work on Distributed Coordination Function (DCF) on 802.11 plus IBSS infrastructure;
- An IBSS is considered to be a Community;
- The collision model is provided by DCF-CSMA/CA protocol;
- The time is assumed to be slotted and each slot has a time duration equal to time required to transmit a packet.

We propose social-based metrics to choose the best node's association. The main idea is to use the knowledge obtained from interactions of each node and its surrounding neighbors, through the concepts of communities, density/quality of the friends and 'friendship' of nodes. We proposed three complementary technical metrics: (i) *Neighborhood Nodes Friendship* (ii) *Associated Nodes Friendship* and (iii) *Community Nodes Friendship*. These metrics, described in the following Sections 3.4.3.1, 3.4.3.2 and 3.4.3.3, will be jointly grouped in a social-based association metric.

3.4.3.1 Neighborhood Nodes Friendship

The purpose of this metric is to quantify the degree of 'friendship' between a node i and its surrounding neighbors ($\mathscr{P}_i$). The node i with more nodes ('friends') in its neighborhood will get an higher 'friendship' indicator. The degree of 'friendship' between any two nodes is given by:

$$\mathcal{F}_{i,j} = \frac{b_{i,j}}{\sum_{k:n_k \in \mathscr{P}_i} b_{i,k}} + \frac{|\mathscr{V}_j| + 1}{|\mathscr{P}_i| + 1} \tag{3.10}$$

where $b_{i,j}$ represents the number of beacons received by node i from node j, and $|\mathcal{V}_i|$ represents the cardinality of the set of neighbors of node i, i.e., the size of the neighborhood of node i. The overall 'friendship' of a node i can then be inferred averaging the 'friendship' between node i and all other nodes in its neighborhood ($\mathscr{P}_i$):

$$\overline{\mathcal{F}_i} = \frac{1}{|\mathscr{P}_i|} \sum_{j:n_j \in \mathscr{P}_i} \mathcal{F}_{i,j}. \tag{3.11}$$

3.4.3.2 Associated Nodes Friendship

The purpose of this metric is to quantify the quality of 'friendship' between a node i and node j to which it is associated: this is a 1 to 1 analysis that will take into account the quality of the association. The 'friendship' quality between node i and node j with an active association is given by:

$$\mathcal{G}_{i,j} = \rho_{i,j} + \mathcal{L}_{i,j} + \mathcal{R}_j \tag{3.12}$$

where $\rho_{i,j}$ is an estimation of the Signal to Noise Ratio between nodes i and j normalized in [0, 1], $\mathcal{L}_{i,j}$ is an estimation of the link stability between the nodes (defined below), and $\mathcal{R}_j$ is a metric that quantifies node j available resources characteristics (such as processor, memory, storage, battery remaining and type of interfaces). This value is normalized in [0, 1] and is transmitted in the beacons between nodes.

The link stability $\mathcal{L}_{i,j}$ is estimated by the ratio between the number of beacons received in node j by node i ($b_{i,j}$) and the total number of beacons sent by node j (s_j):

$$\mathcal{L}_{i,j} = \frac{b_{i,j}}{s_j}. \tag{3.13}$$

The overall 'friendship' quality of a node i can then be inferred averaging the 'friendship' quality between node i and all nodes associated ($\mathscr{A}_i$):

$$\overline{\mathcal{G}_i} = \frac{1}{|\mathscr{A}_i|} \sum_{j:n_j \in \mathscr{A}_i} \mathcal{G}_{i,j}. \tag{3.14}$$

where $|\mathscr{A}_i|$ represents the cardinality of the set of node i associated nodes, i.e., the number of nodes associated with node i.

3.4.3.3 Community Nodes Friendship

This metric quantifies the quality of 'friendship' between a node i and the nodes in the same community. The idea is combine the size of the community (given by the distributed algorithm) together with the average overall 'friendship' quality of all nodes in a community:

$$\mathcal{H}_i = |\mathscr{C}_i| + \left(\frac{1}{|\mathscr{C}_i|} \cdot \sum_{j:j \in \mathscr{C}_i} \overline{\mathcal{G}_j} \right) \tag{3.15}$$

where $|\mathscr{C}_i|$ represents the cardinality of the set of nodes in the same community as node i, i.e., the number of nodes in the same community of node i.

3.4.3.4 Global Social Metric

The global social metric is obtained by performing a weighted sum of the three partial node 'friendship' metrics (neighborhood, associated and community). Defining w_F, w_G and w_H as the weights of the neighborhood, associated and community nodes 'friendship' metrics, respectively, it is possible to define the global social metric as:

$$\mathcal{S}_i = w_F\overline{\mathcal{F}_i} + w_G\overline{\mathcal{G}_i} + w_H\mathcal{H}_i \tag{3.16}$$

where $w_F + w_G + w_H = 1$. The influence of each metric in the quality of the associations and data dissemination is presented in the next Subsection 3.4.5.

3.4.4 Social-Based Adaptive Beacons

Similarly to presented in Equation 3.1 from Section 3.2, we also propose an adaptive beaconing scheme that dynamically adapts the beacon interval according to the conditions of each node. We consider that lower density of nodes and larger communities require faster spread of information: the interval between beacons should be proportional to the neighborhood 'friendship' metric and inversely proportional to the community 'friendship' metric. Therefore, the Beacon interval of node i ($\mathscr{B}_i$) is given by:

$$\lim_{\mathscr{L}\to\mathscr{U}} \mathscr{B}_i = \delta\left(\frac{\overline{\mathcal{F}_i}}{\mathcal{H}_i}\right) \tag{3.17}$$

where δ is a tuning factor. ($\mathscr{B}_i$) is the interval time which can increase or decrease according to the nodes density or by community size without crossing the limits $\mathscr{L}$ lower and $\mathscr{U}$ upper, which vary between 0.1 to maximum of 1 second.

3.4.5 Social Metrics Results

This Section presents a simulation study that evaluates the proposed metrics and its influence in the nodes' association process, their quality and the characteristics of the communities being formed.

3.4.5.1 Ad-hoc Communication Model

The implementation of the mechanisms described in this Section are conducted in Network Simulator v.3.9. We implement the communication process and interaction at MAC layer in order to provide the proper base for ad-hoc connection and interaction in wireless scenarios. In the beginning we found limitations in the standard ad-hoc communication model at the IEEE 802.11 MAC of the NS-3 simulator, thus, reinforcing the necessity to re-implement this model in order to provide: (i) ability to communicate between the nodes without to use centralized entities (e.g. APs); (ii) transmission of broadcast Beacons at regular intervals; (iii) a selector function which selects the scanning type (active or passive); (iv) beacon missed function, which detects the value of the maximum beacons missed and automatically starts a new association; and (v) the association based on the social metrics. The ad-hoc model implemented in NS-3 simulator is a Finite State Machine (FSM) model (see Figure 3.34), which is composed by the bootstrapping, beacon missed functions, scanning mode (active/passive), probe request and response, beacons, association request and response and disassociation.

This FSM implemented serves as the basis for the communication between the wireless nodes. Besides the limitations found on the ad-hoc model of the IEEE 802.11 at the MAC Layer in the NS-3 simulator, NS-3 is still the best choice ensuring that the simulations are close to the real ones. More detail about the implementation modules and interaction at the IEEE 802.11 MAC can be found in [Gom12].

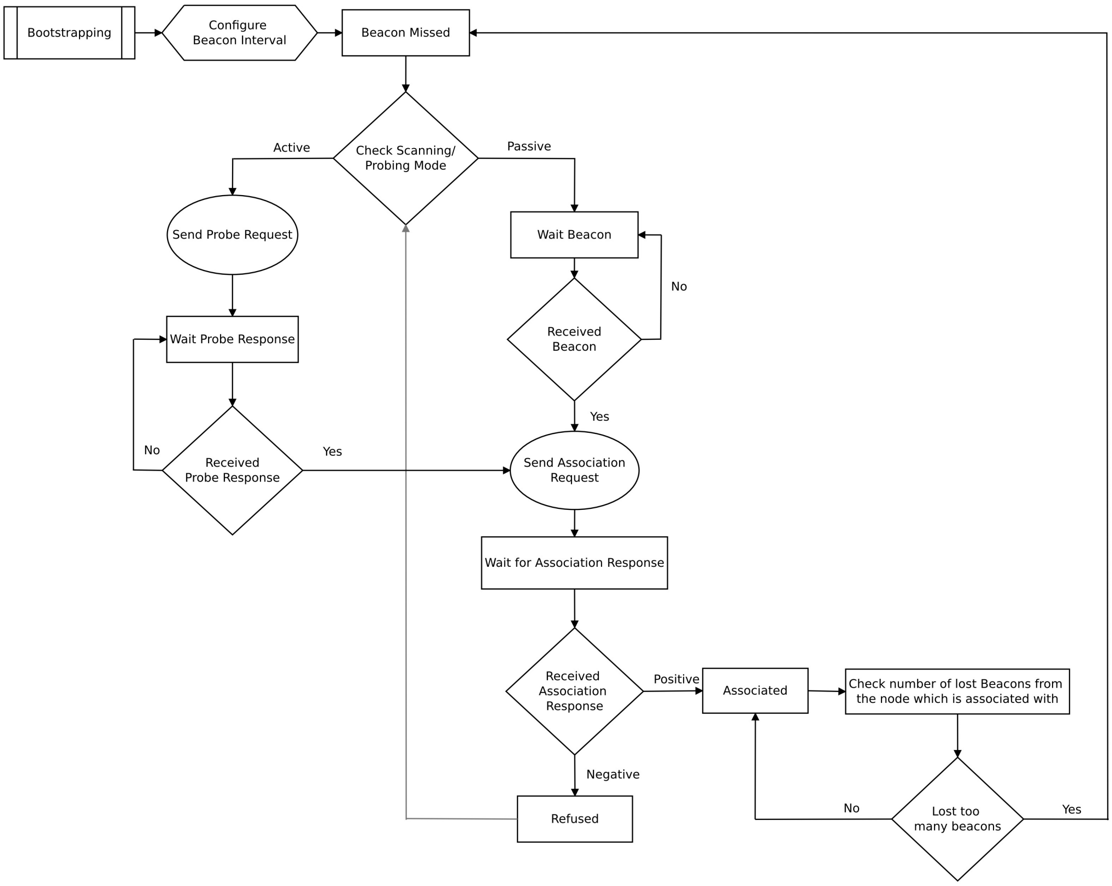

Figure 3.34: FSM implemented on ad-hoc node

3.4.5.2 Nodes Association

The scenario used to assess the nodes association evaluation is modeled by various input parameters that are shown in Table 3.4. The nodes are randomly included in a maximum area of 1000m by 1000m. The obtained results include the mean values of the results in independent repetitions with different seeds, and with confidence intervals of 90%.

Regarding the mobile pattern, *Random Walk* produces random mobility traces without any pattern, thus, conditions will vary quickly causing associations to be made and removed often.

The baseline approach to compare the social-based metrics is the association to the node that sends the first beacon. In this criterion, the nodes are able to associate with a neighbor or community as long as they are in the coverage area, and receives the first contact beacon for the association, which is represented by *(First Beacon)*.

The Figure 3.35 shows a comparison between the two possible ways of scanning: active

Table 3.4: Parameters

Parameters	Values
Simulation time	300s
Initial allocation of nodes	On grid
Mobility pattern	random walk
Initial separation between nodes	50 meters
Number of nodes	4, 9, 16, 25, 36, 49, 64, 81 or 100
Simulation window	proportional to the number of nodes (4 nodes = 40m, 9 nodes = 90m ... 100 nodes = 1000m)
Scanning mode	active or passive
Weights of social metrics	w_F, w_G and w_H

and passive, implemented according to the IEEE 802.11 standard. The average time to regain the association is about 0.04s in passive mode. Using the active mode, its time is around zero in simulations up to 64 nodes. In simulations with 81 and 100 nodes, the active scanning continues to provide faster associations. One factor that contributes to the initial difference is that a smaller density of beacons provided through active scanning introduces fewer collisions, and therefore, the node in active mode can regain association almost immediately after request.

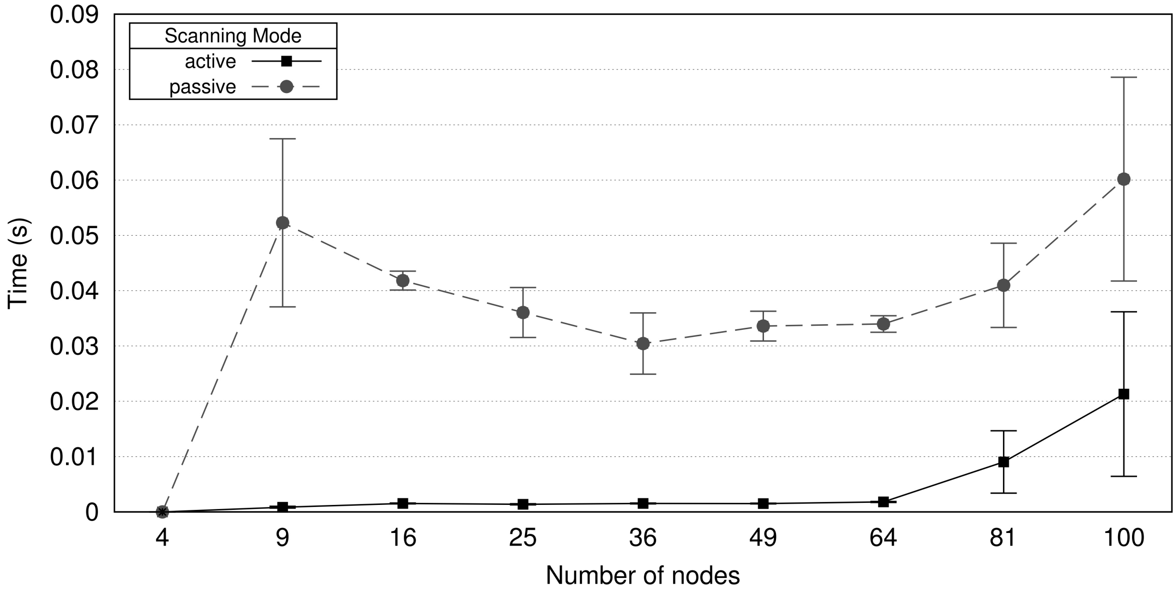

Figure 3.35: Evolution of the time for reassociation

The results presented in the Figure 3.36 evaluate the time to establish the first association with or without social metrics, with different weights for each metric (for example, $w_F = 1$, $w_G = 0$ and $w_H = 0$, and any other combination as seen in the legend). As referred, the baseline comparison is the association through the first beacon received: *(First Beacon)*. In order to clarify the information presented in Figures 3.36, 3.38 and 3.39, we consider in this evaluation the weights of the social-based metrics for the $\overline{\mathcal{F}_i}$, $\overline{\mathcal{G}_i}$ and $\mathcal{H}_i$ to be enabled '1' or disabled '0', or with balanced influence of 1/3 for each weight in comparison to the first contact beacon association. For example, in the Figure 3.36, the weights configured as

$w_F = 1$, $w_G = 0$ and $w_H = 0$ mean that only the factor $\overline{\mathcal{F}_i}$ is enabled in this analysis, similarly with the other factors.

The results show that the time required for all nodes to establish the first association is not changed through the social metrics approach. The aim of this analysis is to prove that the association with any weight combinations of the social metrics does not spend or need any extra time to calculate the first association.

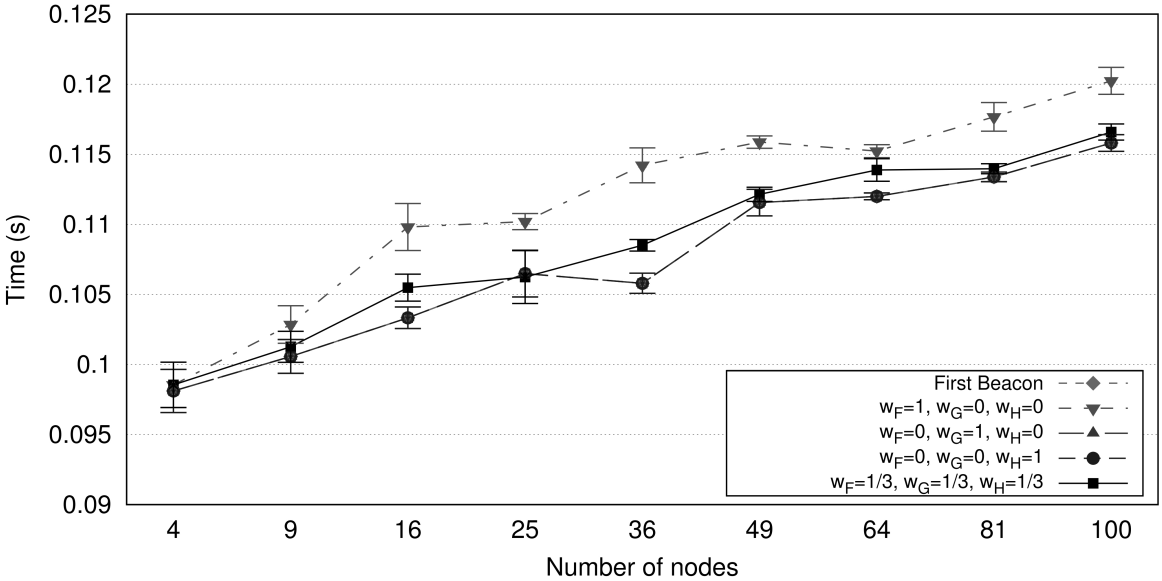

Figure 3.36: Evolution of the time required for all nodes to establish their first association

Figure 3.37 shows the average number of required management packets sent by each node during the simulation, with the number of association responses on the left side, and the number of beacons on the right side. With social metrics (balanced influence of 1/3 for each weight is used), the number of beacons sent is significantly lower, manly justified by the proposed adaptive beacon interval. On the other hand, it is also important to consider that the threshold for the association changes due to a higher social metric, which will have a strong contribution to the total number of associations performed. This factor approaches or separates the curves *Assoc. Resp.* according to its higher or lower values, respectively. The number of association changes is directly related to the threshold value (+10%) configured to enable a new reassociation. Moreover, the number of forced associations caused by social metrics aims to improve the interconnection of the nodes at the MAC layer. In addition, it is observed that the social metrics enforce the nodes to associate to the best neighbors, contributing to communities with better quality according to the desired weight on each social metric, which can enhance the performance of the network at the IP level.

Figure 3.38 shows the number of communities that are present at the end of the simulation. As can be shown in the figure, the nodes tend to glue in a single community, regardless of the number of nodes in simulation. This glue behavior is expected since all nodes establish their association link to the node with higher social metric in their vicinity, creating a decentralized quality community. We can conclude that $\overline{\mathcal{F}_i}$ ($w_F = 1$) exerts a weak influence at this level, resembling the behavior of *(First Beacon)*. In addition, the number of communities is mostly imposed by $\mathcal{H}_i$ ($w_H = 1$) metric. With regard to *(First Beacon)* association, it tends to split in more communities, raising the overall complexity in terms of decentralized management,

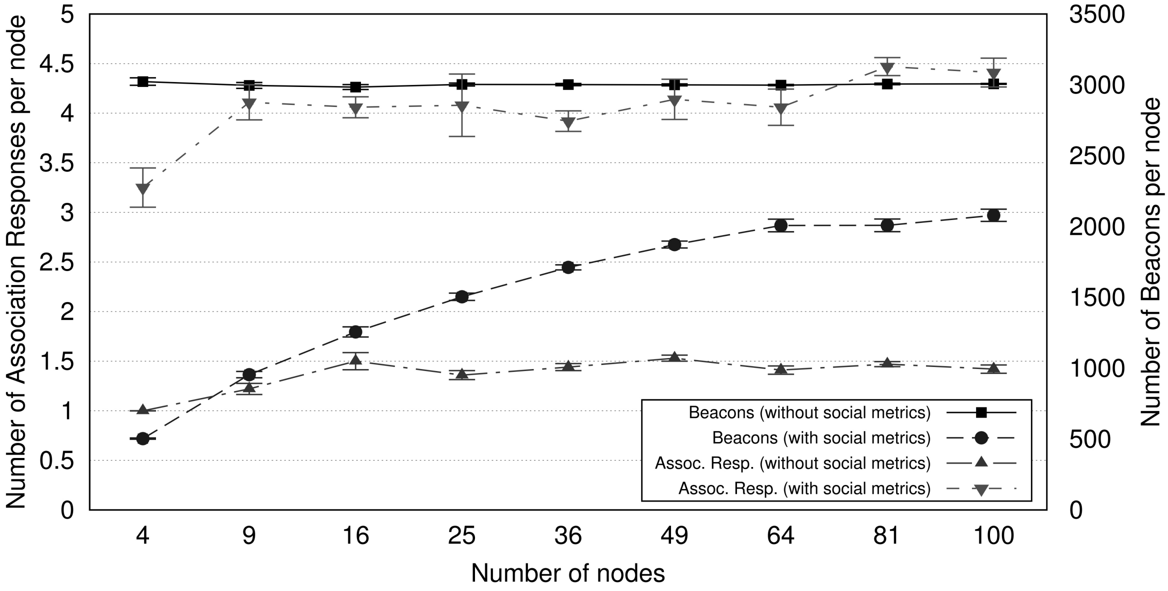

Figure 3.37: Evolution of the number of management packets sent

but reducing the complexity inside each community.

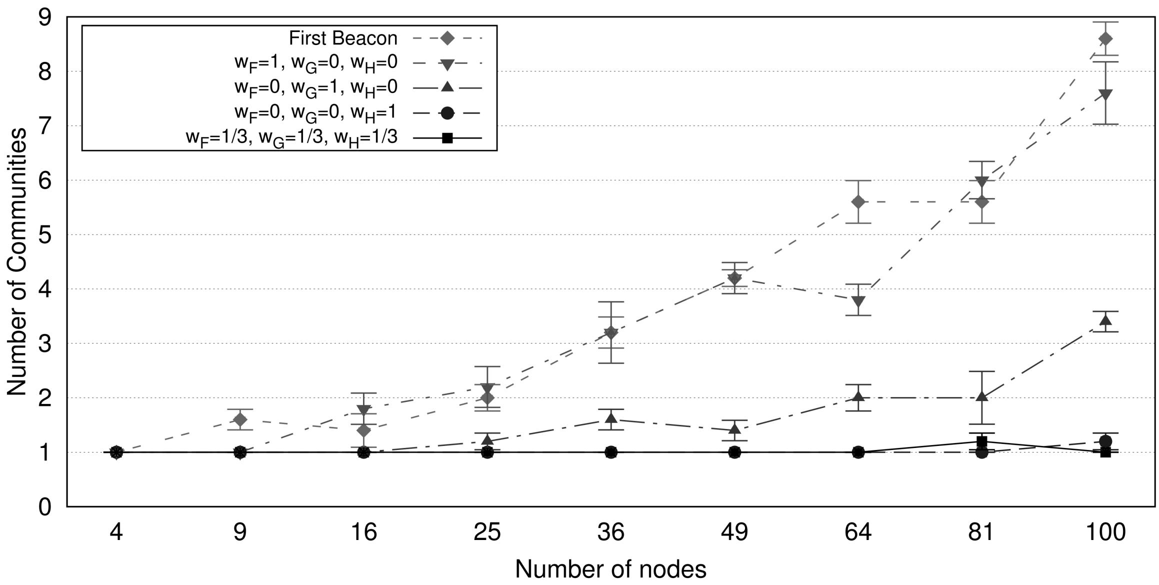

Figure 3.38: Evolution of number of final communities

Figure 3.39 shows the average size of the existing communities until the end of the simulation. This graph is complementary to the previous one (Figure 3.38) and it shows the same trend and influence of each metric.

These results show the benefits provided by our social-based metrics in terms of time to association and reassociation for the support of decentralized and distributed management. Through the active scanning, the nodes can associate as fast as possible, maintaining the stability and connectivity of the links at the MAC layer. Without any extra time to cal-

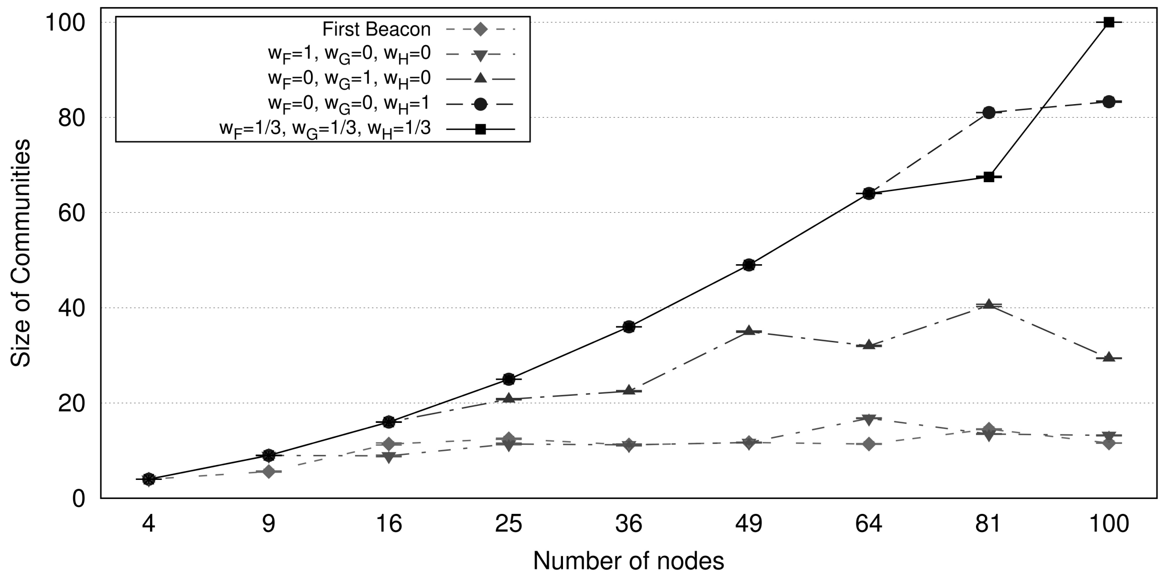

Figure 3.39: Evolution of the size of final communities

culate the first nodes association, the social metrics can also reduce control traffic present at the management packets sent by each node during the communication. Notice that this is extremely important to reduce the required management information exchange between nodes in a decentralized management approach. Moreover, this social metrics approach is extremely useful for scenarios where the high degree of mobility is the main requirement. The social metrics enforce the nodes to associate to the best neighbor nodes as well, contributing to improve the communities quality aggregations and reducing the required reassociations. This enforces a higher stability on the associations and a higher efficiency in the distributed management process.

3.4.5.3 Information Disseminated

In order to analyze how the social metrics affect the network in terms of information dissemination, single 802.11 MAC Data information packets are exchanged between the nodes within the several communities. The dissemination of these packets is performed by assigning a random node to start sending packets; some nodes will receive and forward them according to the dissemination mode, considering two different modes to disseminate 802.11 MAC Data packets: *Pure broadcast* and *Hop-by-hop*. In *Pure broadcast* the nodes broadcast all the received data packets without any criteria and without the need for an established association process. This is the approach used as a comparison baseline reference. The *Hop-by-hop* creates an association path for dissemination, i.e., each node transmits the information by *unicast*, being the destination identified by the node where the association link has been established. In addition, we distinguish two types of association: *First Beacon* and *Social Metrics*. The first one is the default association mode of 802.11 MAC standard, where the criterion used is to establish the association link to the node from where the first Beacon came. The second is the social-based association model, where the criteria used is to establish the association link to the node with the higher social metric, and to constantly monitor the vicinity searching for higher social metrics. Additionally, both association modes are implemented in the 802.11 MAC layer. More detail about different evaluations in terms of time to dissemination and average network usage through social metrics is presented in the Annex C.

With regard to mobile patterns, in *Random Waypoint* nodes travel to specific areas around an interest point with attraction capabilities (e.g points of interest in cities). *Nomadic* induces nodes to choose positions within an area of a mobile reference point, where nodes travel aggregated, similarly to a guided tour (e.g, guided tour in museums, etc). *Random Walk* was already introduced in the previous Subsection 3.4.5.2. These mobile patterns are generated by the software tool *BonnMotion* [Bon] and exported in a format compatible with NS-3.

The scenario is configurable, and those are the main settings:

- 100 nodes in a simulation window of 1000m x 1000m;
- Interval between dissemination flows exponentially distributed with average 1 second;
- *Random walk*, *Random Waypoint* and *Nomadic* mobility patterns with similar settings (e.g. speed of the nodes between [5, 10] m/s);
- *Pure broadcast* and *hop-by-hop* dissemination modes;
- *First Beacon* and *Social Metrics* association modes;
- The weights w_F, w_G and w_H for the social metric results is determined by the experimental combinations;
- Mechanism for loops detection and removal;
- The obtained results include the mean values of the results from independent repetitions with confidence intervals of 95%.

Figure 3.40 shows the percentage of nodes to which the dissemination information has been received. Regarding to the *Random Walk* scenario, we observe a low percentage in all three association and dissemination modes. This is due to the fact that nodes are all apart from each other and do not converge to common point(s), i.e., the movement between several nodes is independent and random in terms of direction and speed. In the other two mobility patterns, (*Random Walk* and *Nomadic*), and considering the social metrics association mode, the percentages are higher in the scenario with fixed reference points (*Random Waypoint*) rather than those where the reference points are mobile (*Nomadic*).

In a scenario where the convergence points are dynamic, the links between the several nodes are harder to maintain than in a scenario where those points are constant, therefore implying lower dissemination percentage achieved. Hence, the association links must be established accurately rather that randomly, and this is the role played by the social metrics, abstracting the concept of the best association and allowing a measurable evaluation of the current conditions of the node in social terms.

Thus, it would be expected that the *Pure Broadcast* dissemination mode reached, at least, the same percentage as the other two methods. However, we must keep in mind that these simulations were performed with 100 nodes, all sharing the same medium, being the access monitored by the Distributed Coordination Function (DCF). When a broadcast dissemination method is used, it is likely that more collisions occur and that the buffers fill more quickly, causing packet dropping. All these effects will have a negative impact on the results achieved. Unlike the *Pure Broadcast*, the other two approaches (*First Beacon* and *Social Metrics*) use the *hop-by-hop* as dissemination mode to forward data packets, thus becoming more efficient and achieving better percentages.

Note that, for this analysis we consider the technical factors measured through social metrics, and as observed in the Figure 3.40, the maximum percentage of nodes that received

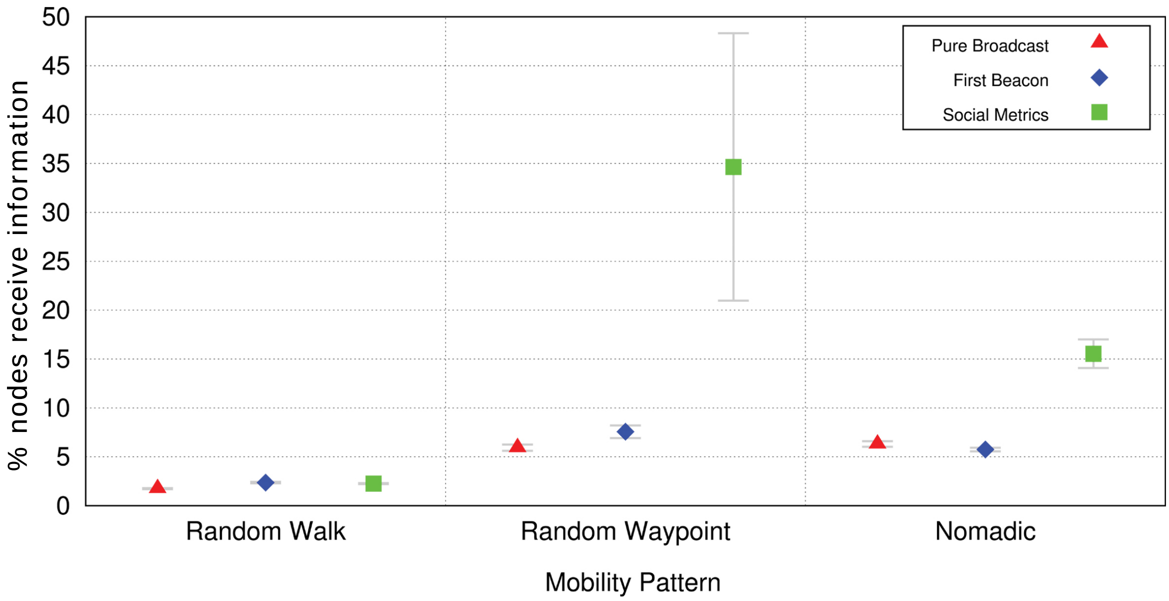

Figure 3.40: Percentage of nodes that received the information disseminated

the information disseminated is approximately 35% in Random Waypoint scenarios. Thus, this result re-inforces the importance of having a well-defined optimization method in order to optimize the social-based weights along with new ways to enhance the dissemination of information in highly mobile scenarios.

3.5 Data Dissemination

This Section is mainly supported by the Papers VII, VIII and IX presented in the Table 1.1 from Chapter 1, and Paper IX is presented in the Annex D.

Cooperative mobile networks are growing every day, especially in the fields of multimedia streaming, content production and dissemination, enabling people to enjoy and share their contents anywhere and anytime. Nowadays, it is common the use of mobile devices by people, in most diverse scenarios. The opportunity to watch a movie, play games, share photos and videos, and watch multimedia streaming are interesting examples of mobile personal entertainment.

It is very important to acquire, process and consider the behavior and preferences of the users (e.g. identify the users relative location and movements patterns and/or instabilities, identify the interaction of the them devices, preferred multimedia content, etc...). Moreover, identifying these behaviors and using that information to optimize the network routing will result in the improvement of the underlying multimedia delivery service. With the human behavior information, networks are able to adapt and re-adapt in order to maximize the quality of service and experience effectiveness of the users. Thus, networks based on human-centric concepts together with the underlying technical characteristics enable cooperative mobile networks, where it is possible to efficiently disseminate multimedia streaming anywhere and anytime.

There are many scenarios where people are able to enjoy multimedia contents, but the requirements imposed by these are very hard to meet, namely person movement instability, person location and prediction, available infrastructures, physical indoor or outdoor barriers, etc. The trend nowadays has changed most networks that are closer to the user from central and hierarchical static network infrastructures to dynamic Device-to-Device (DtD) communications. Considering mobile devices as the main medium to deliver multimedia content to an ever-growing demand for multimedia entertainment, many innovations need to be proposed in order to understand the DtD challenges, behaviors, preferences and interactions. These new demands embrace new services and technologies to allow digital streaming dissemination in mobile devices, offloading those demands from Telecom operators. The mobile applications make it possible to not only have to overcome challenges of system-level services, but also to enhance the quality of service (QoS) and quality of experience (QoE) of the user.

To enable multimedia streaming dissemination over mobile networks, the communication between the devices is done hop-by-hop until reaching the destination devices. Generally, this process is done by mobile routing protocols, gathering the proper information to fulfill their next-hop routing tables. The information gathered is exchanged without distinction in the network layer (e.g. network control information and data multimedia streaming flows), suffering from common problems, such as high end-to-end delays and overhead to maintain the routing tables.

In the previous Section 3.4 we introduced the concept of social-based metrics measuring the nodes and communities quality through technical factors at MAC layer. In order to enhance data dissemination, specially multimedia streaming content, we identify the importance to introduce a new factor which quantifies and qualifies the social preferences of the users in the social network. Those technical factors, presented in the Section (3.4.3.4) through the social metrics and the new social network factor, allow an enhanced inference of the social metrics towards to the DtD behavior. Thus, aiming to create the possibility to maintain the best associations between devices through their behaviors, interactions and affinities. This interaction between technical and social network preferences is described with greater detail in the Annex D.

The new social network factor takes into consideration the information obtained from social networks, identifying and quantifying the amount of human similarities. This metric is inferred by identifying the number of common social network relations between persons (e.g. Facebook, Google+, Myspace, Twitter, etc).

In analytical terms the social network criterion is given by:

$$\mathcal{FF}_{i,j} = \left(\frac{\mathcal{SN}_j + 1}{\mathcal{SN}_T + 1}\right) \tag{3.18}$$

where $\mathcal{SN}_j$ represents the number of social networks the users on node j is friend with the user on node i. $\mathcal{SN}_T$ represents the total number of social networks considered as reference, and +1 avoids division by zero.

Therefore, the final value of the DtD is given by:

$$\mathcal{DTD}_{i,j} = \mathcal{S}_i \times \mathcal{FF}_{i,j} \tag{3.19}$$

where the inferred technical behavior from social-based metrics $\mathcal{S}_i$ (Section 3.4.3.4) from node i can be merged with the social network factor between the user on node i and the user on node j, enabling an enhanced version of the social-based metrics represented by the DtD behavior quantifier between node i and j.

To disseminate multimedia contents, instead of using only technical information at MAC layer, we consider necessary to evaluate the quality experienced by the interested users in the social network as well. Thus, adding the social network behavior experienced by the criterion $\mathcal{FF}_{i,j}$, the users that belong to the same social groups will have higher probability to share the same content, enforcing the nodes affinity at the network level.

In the next Section 3.5.1, we show the influence of DtD behavior, demonstrating the efficiency and effectiveness for the dissemination of multimedia streaming with optimized weights of the social-based metrics $\mathcal{S}_i$ (Section 3.4.3.4) through Genetic Algorithm (GA).

3.5.1 Genetic Algorithm: Solution Approach

In this Section we present a GA approach to optimize the input weights presented in the previous Section 3.4.3.4 to enhance the association and communication between the nodes. We present a GA heuristic to determine the input combination of values that lead to the best solution. Therefore, a payoff function to guide the direction of the optimization process, i.e., to determine "how good" is each combination of inputs is designed. This approach is applied to network scenarios with and without mobility, where the links between nodes are established according to dynamic social factors, such as the social environment, friendship between users and also technical limitations of mobile devices. These factors are merged into an overall weighted formula that is directly influenced by the input parameters.

The network scenario comprises nodes with or without mobility, which are randomly allocated and interact with each other. These interactions can be done directly if nodes are within the predefined coverage range, or indirectly using information forwarded by others. The results of these considerations are briefly illustrated in Figure 3.41. In *t=0* it is shown all possible links that can be performed between nodes. In *t=1* it is represented the established links that lead to the best network performance, according to some optimization criteria. In *t=2*, due to the mobility of nodes and their limited coverage range, the group is divided into two groups. Besides the dynamics of the network, the weights defined by the optimization process should be able to continuously determine which are the links that should be established in order to improve network performance.

The final goal of the proposed mechanism is to be able to determine the input weights that best fit our criteria, according to the following input restrictions:

- There are three different input weights (w_F, w_G and w_H);
- The value of each one can vary between $[0.0; 1.0]$, representing its percentage;
- The inputs are directly proportional: $\sum_{i=1}^{3} w_i = 1$.

The main reasons that distinguish GAs from other optimization methods are summarized in [Gol89]: the set of input parameters is encoded and not directly manipulated; they search from a set of points rather than a single one, allowing the improvement of its convergence; a payoff (fitness) is used to evaluate each solution and not derivative functions or other auxiliary knowledge, turning it into an easy-to-use black-box; and the use of probabilistic instead of deterministic rules, which reduces the probability of being trapped in a local optimum. From conceptual point of view, GAs can be seen as a black-box that requires a set of input parameters and, taking into account intermediate evaluations provided by the fitness function, it is able to provide the best solution. This simplicity of use, together with the other presented reasons, turn the GAs into an attractive optimization tool.

Herewith, GA is used to optimize the performed experiments in NS-3 simulator. In Figure 3.42 it is represented the interaction between those nodes. The GA interacts with NS-3 using multiprocessing, requesting simultaneous simulations with different input values (weights w_F, w_G and w_H). Next, the output of each experiment is analyzed in order to evaluate the result in terms of dissemination percentage p and distance d, where p is the dissemination percentage [0:1] and d is the distance to the information source node. The value of p in the condition is the one that prevents the fitness function to return negative values. These will be

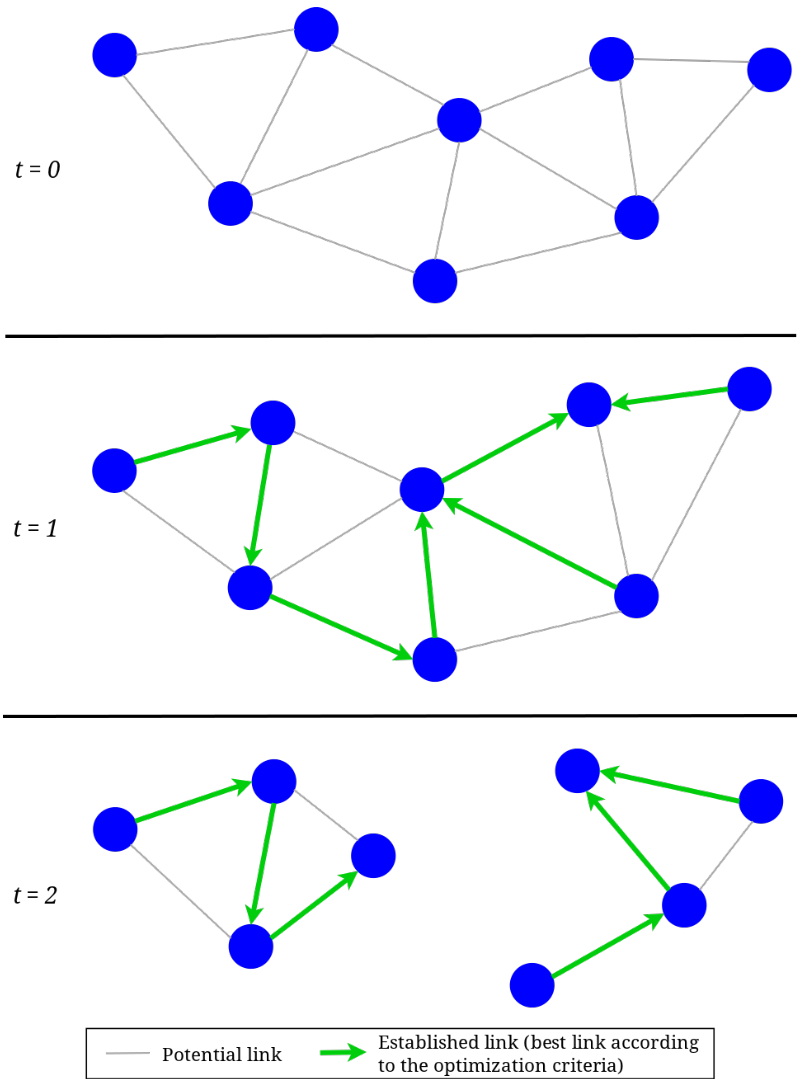

Figure 3.41: Optimization performed from the network perspective

the inputs to the fitness function, which is responsible to determine the corresponding fitness value of each experiment (i.e. of each combination of weights). In the end of each experiment, it is added to a pool that will lead to a new generation. The higher the experiment fitness, the higher the probability of the corresponding combination of weights to be selected to the next generation. As can be seen, all elements are interdependent and tightly coupled.

However, the value of the selected inputs w_F, w_G and w_H cannot be directly manipulated, but rather they must be encoded into chromosomes. Several schemes can be used (binary, permutation, tree, etc.), being the binary the chosen one in this solution. Although this coding scheme cannot be directly applied to many problems, using enough bits for precision and a simple conversion from the natural system units (integer, float, etc) to binary should be sufficient to use the scheme. In the presented problem, knowing that the input weights w_F, w_G and w_H can vary between 0 and 100%: $\sum_{i=1}^{3} w_i = 1$, the chromosomes have been designed with 21 bits. Therefore, each weight takes 7 bits, so that there are enough bits to meet the precision requirements ($\frac{max\%}{2^{bits}} = \frac{100}{2^7} \approx 0.8\%$, thus it is possible to vary each weight in steps of 0.8%).

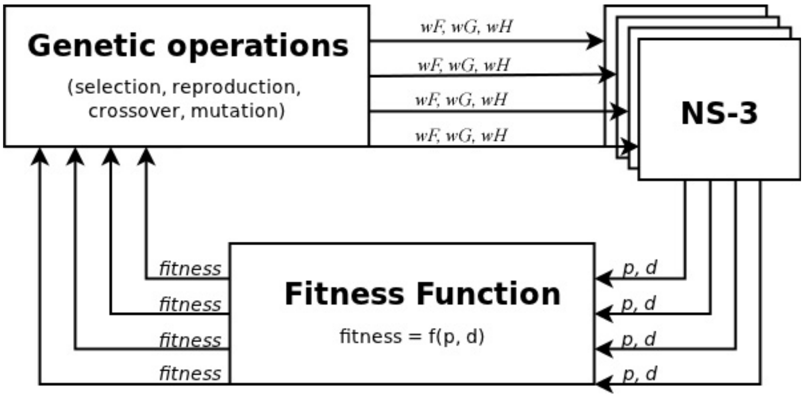

Figure 3.42: Interaction between Genetic Algorithm and NS-3

Taking into account the required number of bits for the chromosomes and other parameters related to the GA behavior, the optimization method is tuned to our solution (Table 3.5).

Parameter	Value
Chromosome bits	21
Population size	20
Max generation	100
Mutation prob.	0.1
Crossover prob.	1

Table 3.5: Genetic algorithm parameters

Although the genetic operators have an important role in the optimization evolution, the element which indicates the degree of optimization is the fitness function. Therefore, it is crucial that it is designed to describe the objective that it aims to model with very high accuracy. Specifically to this problem, it is known that the goal is to maximize the dissemination of nodes, and at the same time minimize the distance to the information source node. Therefore, the fitness function should benefit higher values of dissemination percentage (positive exponent) and negatively affect the hops distance (negative exponent).

This leads to the normalized formula:

$$fitness(p,d) = \begin{cases} \frac{100 * e^{p} + (-e^{d})}{100 * e^{1} + (-e^{0})} & \text{, if p} > \ln\left(\frac{e^{d}}{100}\right) \\ 0 & \text{, if } p \leq \ln\left(\frac{e^{d}}{100}\right) \end{cases} \quad (3.20)$$

In Figure 3.43 it is represented the tridimensional plot of the designed fitness function. It can be seen that, according to the assumptions previously made, higher dissemination percentages result in a higher fitness value. On the other hand, as the distance to achieve that same percentage grows, the fitness value is gradually reduced until a threshold is reached, where the fitness drops exponentially.

3.5.2 Device-to-Device Dissemination Results

This Section evaluates the proposed DtD solution with optimized weights through simulation. In this simulation we consider only multimedia traffic between the nodes. At any instant, any node may start a multimedia streaming. The broadcasting nodes and starting time are

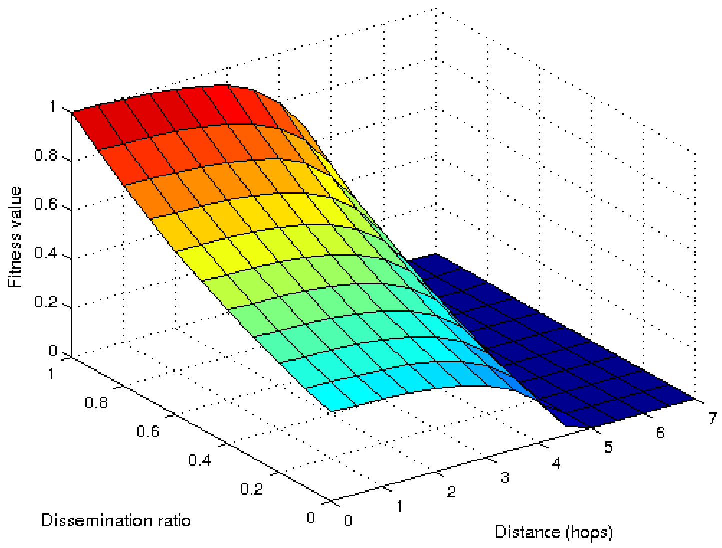

Figure 3.43: Fitness function 3D plot

randomly chosen. The multimedia streams have a rate of 10 packets/s and an exponential random duration, according to the parameters presented in the Table 3.6.

Parameter	Default value
Data rate (pps | bps)	10 | 10240
Packet size (bits)	128
Packet interval (s)	Packet size / Data rate
Max bits	Data rate * Packet size

Table 3.6: Multimedia dissemination flows parameters

Moreover, the users may or may not express interest in the video streaming transmission. We assume that a user associated with a friend at a social network that is generating the multimedia stream has a higher probability (80%) of interest in the content. The stream interest of a node is randomly generated upon the start of a new stream.

The proposed optimization solution through DtD is evaluated considering the DtD:Balanced and DtD:Optimized (optimized by GA). For the DtD:Balanced we consider the weights for the technical factors of the social-based metrics $\mathcal{F}_i$, $\mathcal{G}_i$ and $\mathcal{H}_i$ from the Section 3.4.3 with balanced influence of 1/3 for each weight.

Other nodes association approaches are also compared against these ones: the Standard, where nodes associate to other nodes who first contact (First Beacon) similarly to presented in the Subsections 3.4.5.2 and 3.4.5.3, and keep the association link unless they move apart from each other; the Wireless Spanning Tree Protocol (WSPT) [GGSS13b] that aims to choose the shorter links that lead to the root with the lowest identifier.

In our scenarios we assume a distributed wireless network consisting of nodes that can communicate through peer-to-peer interactions within the limited coverage of each other. Similar to the evaluation of social-based metrics, the simulations for the Device-to-Device data dissemination results are carried out in NS-3 simulator. We consider to present the results of two mobility patterns: *Nomadic* and *Random Waypoint* already described in the

Section 3.4.5. The evaluations and results for different mobility patterns and scenarios is described with greater detail in the Annex D.

The main settings of the setup scenario are described in Table 3.7. The obtained results represent the average values of 10 independent repetitions with confidence intervals of 95%.

We have considered three evaluation metrics: (i) stream reception quality, which quantifies the percentage of successful delivered streaming packets and increases proportionally with the quality of the reception (i.e. higher is better); (ii) distance to stream source, which quantifies the average number of hops to the stream source node, and which can be considered proportional to the transmission delay added to the stream packets (i.e. lower is better); and (iii) average load efficiency ratio which quantifies the total number of multimedia packets divided by the average number of contacted nodes per multimedia flow (i.e. lower is better).

Parameter	Value
Simulation area	150m x 150m
Mobility Pattern	Nomadic and Random Waypoint
Speed	[5, 10] m/s
Number of nodes	10, 20, 30 and 40
Max. consecutive missed Beacons	5
Interval between Beacons	0.1s

Table 3.7: Simulation parameters

Figures 3.44 and 3.45 represent the nomadic scenario. In the first figure, it is possible to observe that using a GA (DtD:Optimized) as an optimization solution, the obtained reception quality can be doubled relatively to the solution with balanced weights (DtD:Balanced). The price to pay is the time for the convergence of the optimization mechanism, i.e., pre-processing time in order to find out the best weights for the specific scenario.

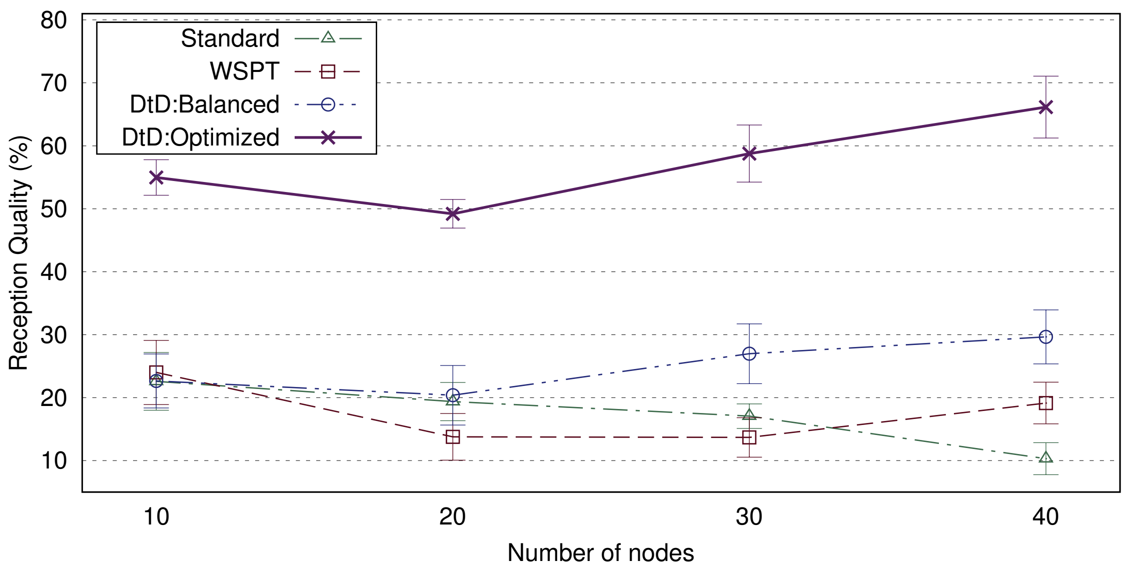

Figure 3.44: Reception quality of information in Nomadic environment

The remaining two solutions, Standard and WSPT, are not able to deal with such dynamic environment, achieving a low percentage of reception quality. The first strategy detects

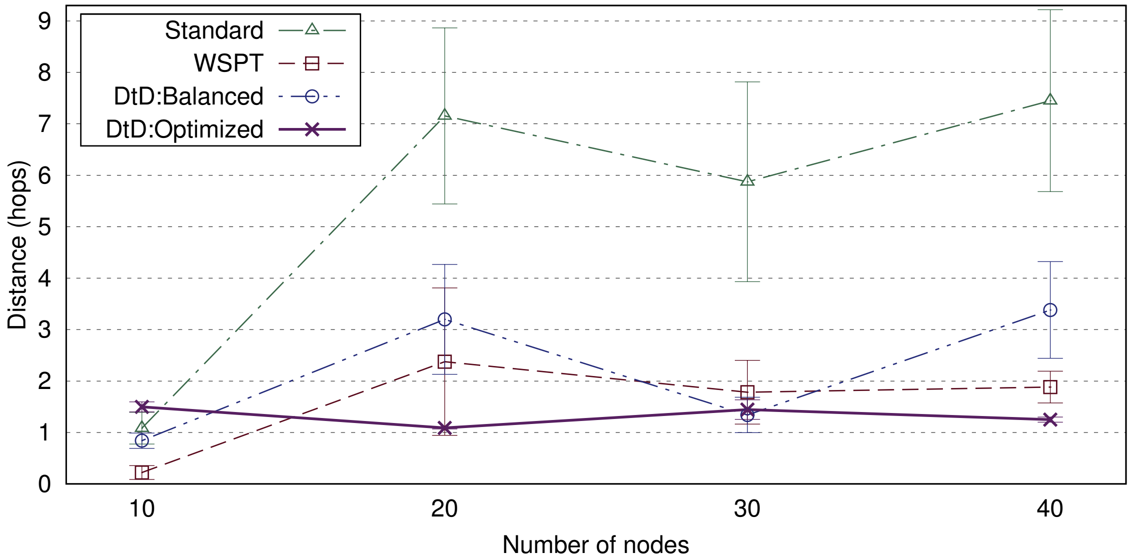

Figure 3.45: Distance to information source in Nomadic environment

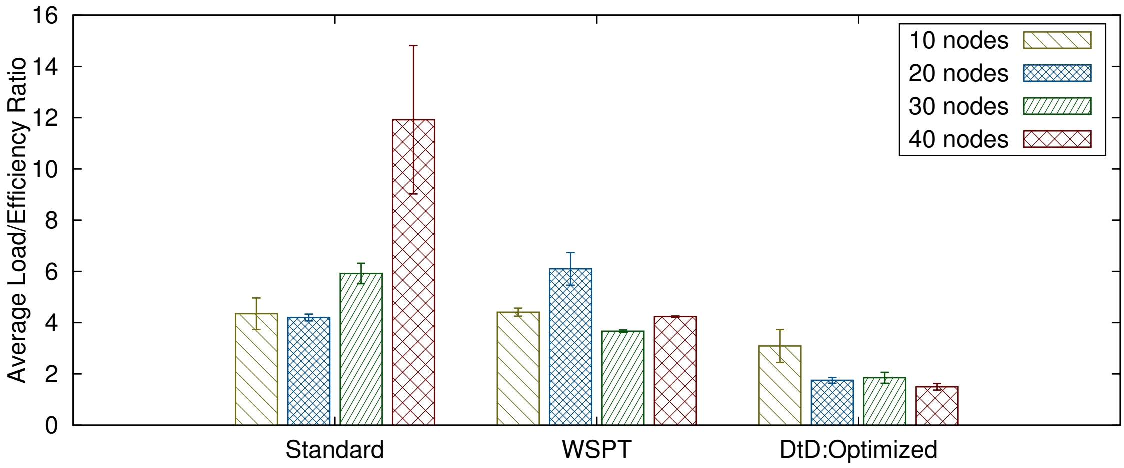

Figure 3.46: Average load efficiency ratio in Nomadic environment

the network changes too late, responding too slowly. The second one tries to maintain the minimum cost path towards the root under constant network topology changes, which means that even the root can change often, temporarily disabling the communication between nodes.

Regarding to the average load efficiency ratio, the DTD:Optimized shortens the distances between the multimedia source to destination, which improves the average load efficiency ratio, as can be seen in the Figure 3.46. In spite of the Standard and WSPT solutions, the average load ratio becomes less efficient than the DTD:Optimized, due to the higher distance (see Figure 3.45) and lower dissemination reception (see Figure 3.44); thus, more non-delivered packets remain into multimedia streaming flows.

Figure 3.45 shows that the optimized solution tends to link nodes directly to the source in order to ensure the minimum distance between nodes and the information source node. Once again, we are assuming that the higher the number of hops to the source, the longer is the delay for the reception of information, because the number of relay nodes is higher.

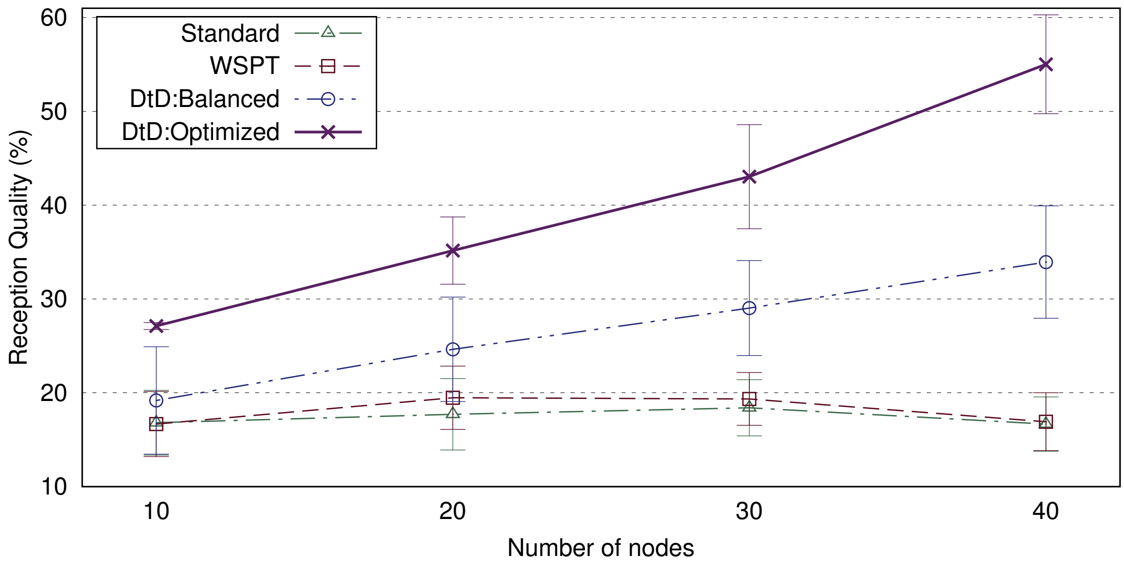

Figure 3.47: Reception quality of information in Random Waypoint environment

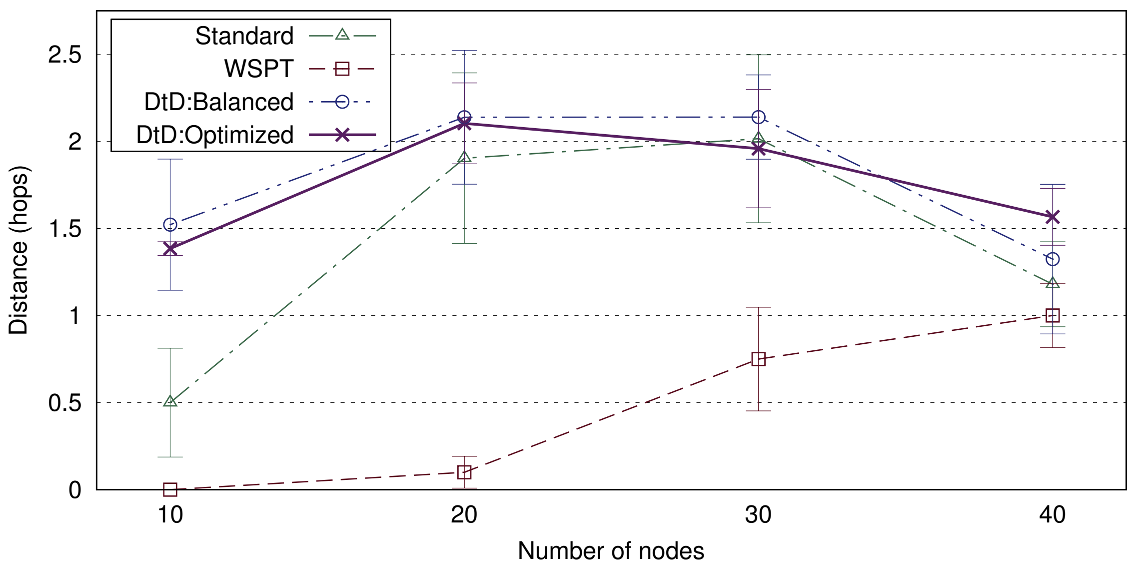

Figure 3.48: Distance to information source in Random Waypoint environment

Figures 3.47 and 3.48 represent the waypoint scenario. In the first figure, the results show that the optimization process can improve the average quality received in about 50% relatively to the same solution that uses balanced weights (DtD:Balanced), almost without affecting the distance to the source (Figure 3.48).

The average load ratio evaluation follows the same behavior of the *Nomadic*, which reduces the amount of multimedia packets per average flow (see Figure 3.49). Note that, even with the larger distance from multimedia source generator (Figure 3.48), the DTD:Optimized

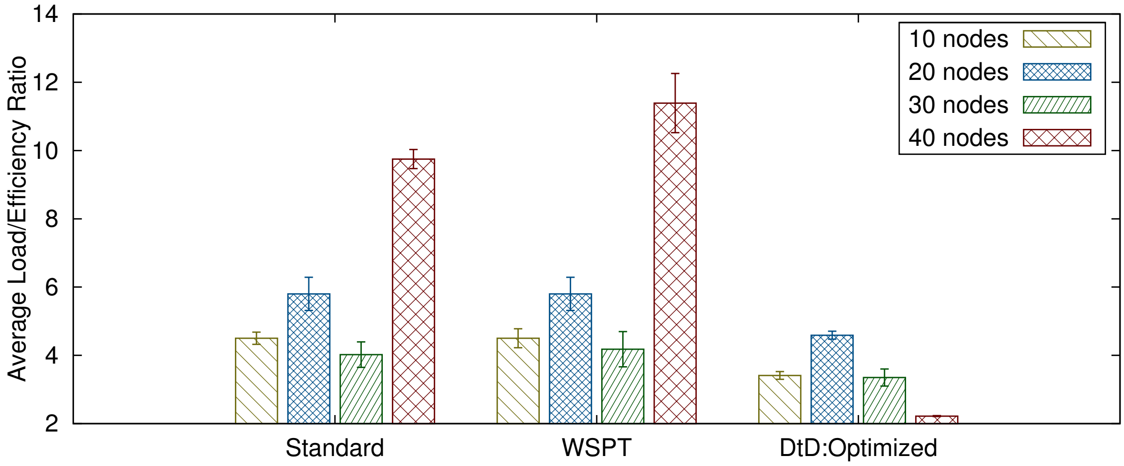

Figure 3.49: Average load efficiency ratio in Random Waypoint mobility environment

compensates the average load ratio by the higher rate of packets delivery, as can be noted in Figure 3.47.

Although the reception quality between WSPT and the Standard approach are nearly the same, the WSPT is always able to establish the shortest possible path (Figure 3.48).

Finally, the objective of the optimization process, modeled by the proposed fitness function, has been achieved in highly dynamic scenarios.

3.6 Proof-of-Concept: Multi-Network Technologies Testbed

From the conceptual point of view, the communication framework integrating the mechanisms proposed in this research work is presented in the Figure 3.1. Herewith, the interaction between the mechanisms for initial communication, exchange of information, network (re) organization and data dissemination were presented. The network (re) organization stage (Section 3.4) through the social-based metrics, and the data dissemination stage (Section 3.5) through the Device-to-Device behaviors are evaluated in NS-3 simulator, therefore not making part of this testbed integration and evaluation as well.

In order to show the feasibility of the initial communication (Section 3.2) and exchange of management information (Section 3.3) stages of the communication framework, we built up a real-world multi-network technologies testbed. The mechanisms that perform this evaluation are HISK2D with the NED function, including functionalities for bootstrapping, discovery, exchange of management information and fault-management working beyond the limit of direct connected nodes.

Additionally, the output information generated from the local repository of each embedded device can be used as input to provide proper information for autonomic decision systems. In order to take network decisions the base information should contain inputs for MAC address of source/destination devices, interfaces, discovered IP addresses from directly connect devices and larger than 1-hop, link bandwidth available and depth of the gathered information, etc. Thus, this base information can be provided by the communication framework in order to promote autonomic making decision processes, such the case of path and bandwidth reservations.

For the testbed (Figure 3.50), we consider scenarios of multiple and concatenated heterogeneous networks both for the backhaul and access networks. The architecture integrates different topologies and technologies. The involved equipment includes: 5 single-boards computer Cambria GW2358-4 using OpenWrt Bleeding Edge, r35830 as operation system; virtual wired grid containing OpenWrt Bleeding Edge r28129 (Guest Xen paravirtualized) as 25 virtual machines on a HP Proliant server (CentOS-5 kernel Xen). Access devices are: 1 EEPC netbook, 1 laptop, 1 Cambria and 1 desktop computer. For long range technology 1 WiMAX IEEE 802.16e Alvarion Extreme 5000 (Base Station) and 1 Customer-Premises Equipment (CPE) are used. Moreover, all devices, excluding the server, remote clients, the WIMAX BS and CPE, perform and process instances of the HISK2D with NED function mechanisms.

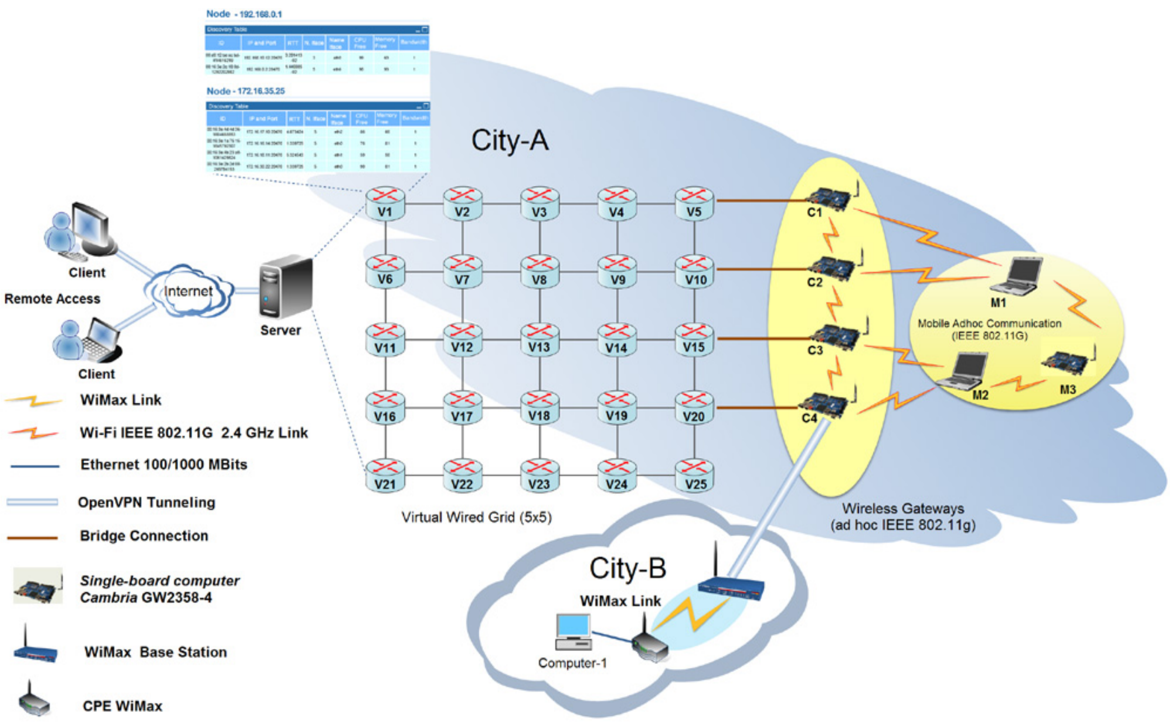

Figure 3.50: Overview wimax and mesh architecture

As can be seen in the Figure 3.50, the client nodes can access the remote PC using a remote desktop application (e.g. TeamViewer) which can manage the testbed functionalities through a web-based interface (e.g., virtual node V1). For the virtual grid topology, the devices are connected through standard IEEE 802.3 network LAN on different sub-nets, according to the device connected interface (e.g., eth0 172.16.0.1, eth1 172.16.1...). The grid topology (similarly to presented in the Sections 3.2 and 3.3) is created through a Python script which automatically generates the virtual bridges and link connections between the virtual machines. The bridging between the virtual border machines (e.g. V5, V10, V15 and V20) to the wireless Cambrias (e.g., C1, C2, C3 and C4) is performed by VLAN connections. With respect to the ad-hoc wireless gateways through standard IEEE 802.11, the wireless Cambrias (e.g., C1, C2, C3 and C4) are used to extend the virtual grid topology in order to share the wireless connections, also to the mobile ad-hoc network devices (e.g. M1, M2 and M3), integrating all different topologies and technologies involved. Additionally, MAC filters are used to filter the communications devices, in order to provide properly ad-hoc connections (e.g., C1 may have to communicate through C2, to reach the device C3).

In addition, the server, the virtual grid topology, the wireless cambrias C1, C2, C3, C4 and the access devices M1, M2 and M3 are situated in the City-A (Aveiro, Portugal), and the WiMAX BS and CPE and the access Computer-1 are located in the city-B (Covilhã, Portugal). In order to unify the network access between both cities, tunneling and virtual private network techniques over a dedicated high speed intranet network are used.

The backhaul of the IEEE 802.11e mesh network is assured by a WiMAX point-to-multipoint IEEE 802.16e Base Station (BS) operating at 5.4 GHz. Furthermore, the nodes explore this remote connection with the BS in order to perform the discovery and exchange of management information functionalities.

3.6.1 Evaluation

The previous experimental results presented in the Sections 3.2.1 and 3.3.2.1 were performed separately on the wired grid scenarios. The purpose of this evaluation is to show the performance of the HISK2D with NED in multiple technologies (e.g. wired and wireless) unified in the testbed. In this analysis, the unification between wired and wireless is concerned, and the performance of the HISK2D with NED is measured in terms of time for link error detection, recovery, propagation and update of the management information, time to discover wireless devices, average CPU overhead and time/number of discovered devices on the worst case scenario (which it means using high background traffic load).

To introduce background traffic, IPERF [ipe13] is used in each device interface, injecting different level of background traffic (e.g. low=10%, medium=50% and High=80% of the bandwidth capacity of the link). The CPU overhead is measured through the UPTIME unix-based command line for every 1 second. Both traffic injection and CPU load have an observation time of 5 minutes. The links capacity differs from wired and wireless scenarios and the software IFTOP [ift13] is used to measure the real bandwidth capacity on the warm-up stage of the experiments, defining the correct traffic load levels. The virtual grid set up is similar to the one presented in the experimentation results in Sections 3.2.1 and 3.3.2.1. The wired grid topology size considered in this integration is (5x5) 25 virtual devices as presented in the Figure 3.50.

In this evaluation, we compare the discovery and management information exchange with well-known wireless discovery baselines, using OpenWrt open-source versions of OLSR version 0.6.3 and BATMAN [bat08] daemon r1439-1 version for wireless. These are the protocols used because the grid is integrating with wireless ad-hoc nodes.

The interval between periodic packets (e.g. HELLO) for the wireless scenario is the default value of the protocols (ORIGINATOR interval = 1 second for BATMAN and HELLO interval = 2 seconds for OLSR), following the standard [bat08], [CJL03] specification. In the HISK2D with NED this interval is dynamically adapted with a mean value around 10s for the wired grid and 5s for the wireless scenario. Moreover, the clocks of all devices in the testbed are synchronized through the Precision Time Protocol daemon (PTPd) [EL02], in order to minimize the offset that occurs among clocks. The values presented in the graphs below are an average of 5 repetitions and 95% of confidence interval.

For this experimentation results, we have considered four evaluation metrics: (i) signaling cost in bytes and in number of packets, both measured for each wireless device (i.e. lower is better); (ii) recover/update average time, which quantifies the time to propagate recover/update messages, after a fault (i.e, node or link) to be detected by the fault-management process (i.e. lower is better); (iii) average CPU overhead, which quantifies in percentage the impact of CPU overhead with or without background traffic injected (i.e. lower is better); and (iv) discovery time (ms) and the number of discovered nodes, which quantify the time to discover the nodes (i.e. lower is better) and the total number of discovered nodes in both wired and wireless scenarios integrated (i.e. higher is better).

Figure 3.51 shows the signaling cost in bytes and in number of packets exchanged in the discovery and information management exchanged/updated for each wireless cambrias and access network devices.

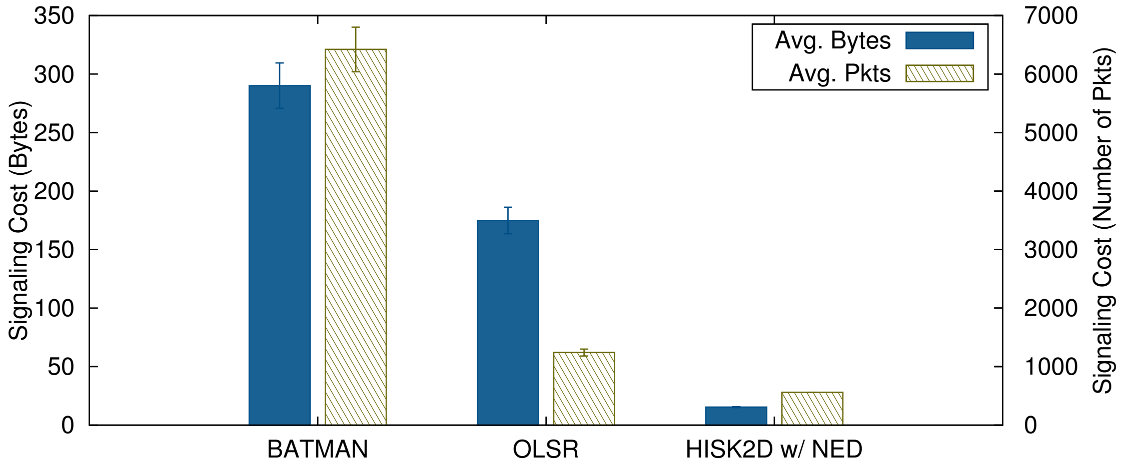

Figure 3.51: Average signaling cost (bytes and number of packets) for each wireless device

As observed, HISK2D with NED has lower signaling cost and requires less bandwidth to control the management information when compared to the BATMAN and OLSR protocols. BATMAN sends excessive Originator Messages (OMG), advertising the existence of devices. Although, OLSR sends several topology control (TC) messages to discover and disseminate link state information. Regarding to discovery and management information exchanged, monitoring, and updating of links, note that, both OLSR and BATMAN have higher signaling cost than HISK2D with NED in the network.

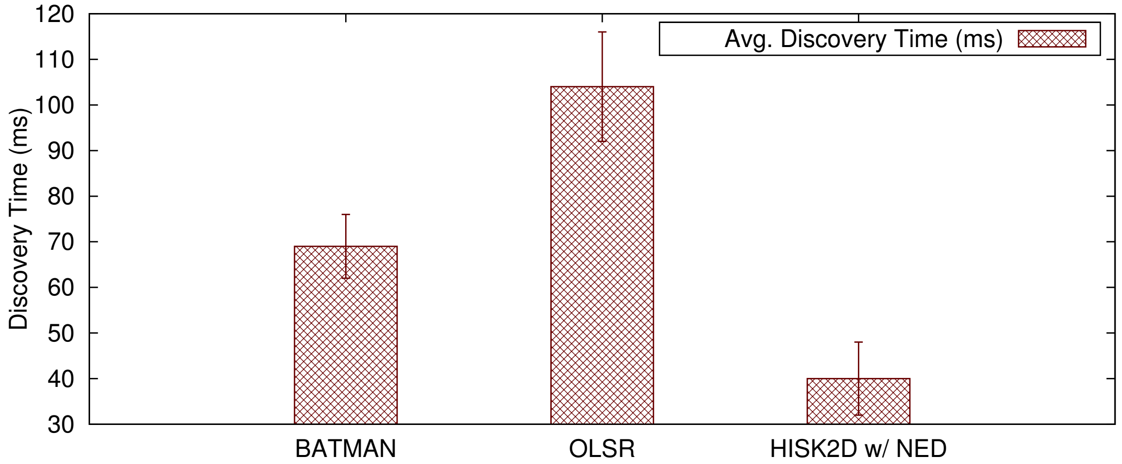

Figure 3.52: Average time to discover the wireless devices

Figure 3.52 shows the average time to discover the wireless devices. HISK2D with NED requires a lower time ($\approx$ 40 ms) to discovery the wireless devices in the testbed scenario (total of 7 wireless devices according to the Figure 3.50). OLSR spends more time to choose the Multi-Point-Relays (MPRs) nodes, which are the nodes that propagate the discovery messages in the network. In the case of BATMAN, it sends constantly discovery messages with 1 second interval, regardless if the network increases or decreases. In case of HISK2D with NED discovery, the initial exchange of discovery messages is configured in the bootstrapping process, and it is initially lower (0.5 seconds) than the adaptive HELLO limit $\mathscr{L}$ presented in Equation 3.1 from Section 3.2. After exchanging a few discovery messages, the adaptive HELLO interval functionality is normalized and HELLO messages are sent without crossing the limits $\mathscr{L} = 1$ second and $\mathscr{U} = 5$ seconds. Thus, HISK2D with NED has clear advantages in terms of average time and number of discovered wireless nodes when compared to the baselines analyzed.

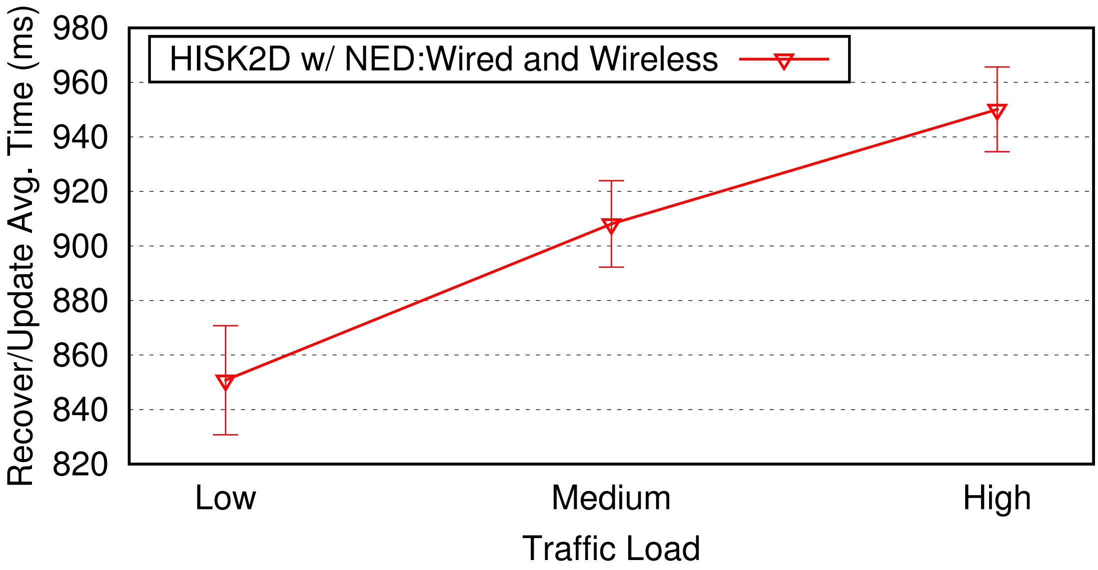

Figure 3.53: Link recover/update time in function of traffic load

The average time to recover/update the management information for any device is measured and it is shown in Figure 3.53, assuming a link error originated from the wired grid devices (e.g. Virtual Node V1, at the extremity of the grid). HISK2D with NED spends approximately 850 (ms) when the background traffic is low, and for the worst case it spends 950 (ms) when traffic load is high. At a first glance, for both wired and wireless scenarios, HISK2D with NED spends less than 1 second to detect, propagate, recover and update the management information caused by a link error in all devices in the network. The depth of the propagation messages is configured at bootstrapping process, and depends on the size of the network at a certain depth (e.g. 10 hops).

The impact of the CPU overhead is also measured in Figure 3.54. Without HISK2D with NED and background traffic processes running, the reference of CPU load is almost 0.061%. Only with background traffic running, the consumption is almost 1%, 2% and 3% for traffic load low, medium and high, respectively. With both HISK2D with NED and background traffic processes running, the CPU overhead increases to 3%, 5% and 6%, respectively. Thus, HISK2D with NED shows to consume lower CPU for executing the functionalities of discovery and exchange of management information, according to the desired depth for both wired and wireless scenarios analyzed.

The average time/number of discovered devices from different depths in worst case scenarios is also quantified in Figure 3.55. For depth equal to 2 hops, the HISK2D with NED spends 210 (ms) to discover approximately 8 devices, whereas for depths equal to 10 hops,

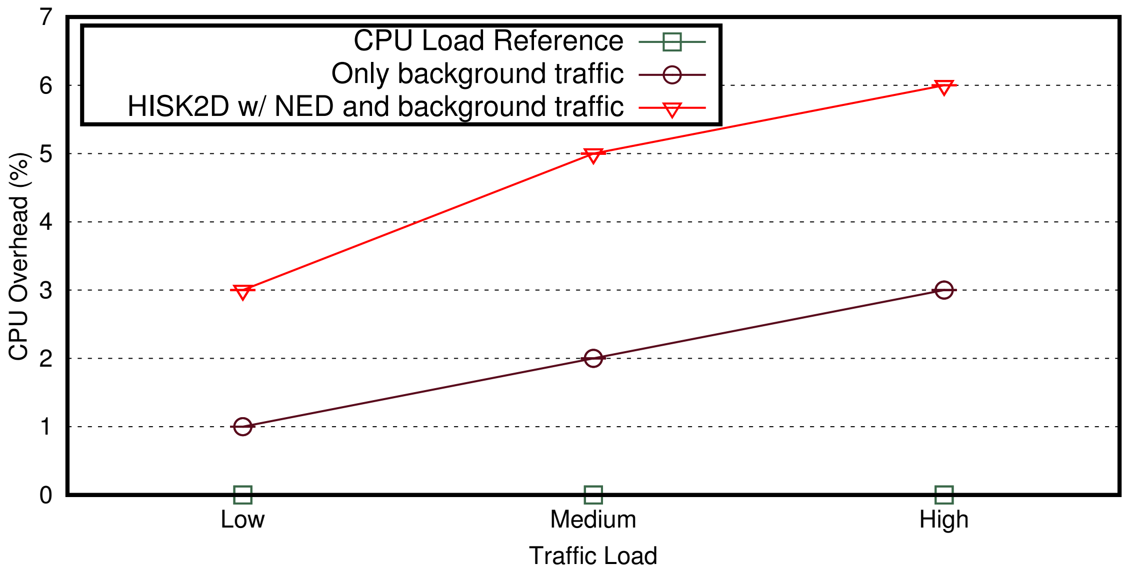

Figure 3.54: Average CPU overhead in function of traffic load on both wired grid and wireless scenarios

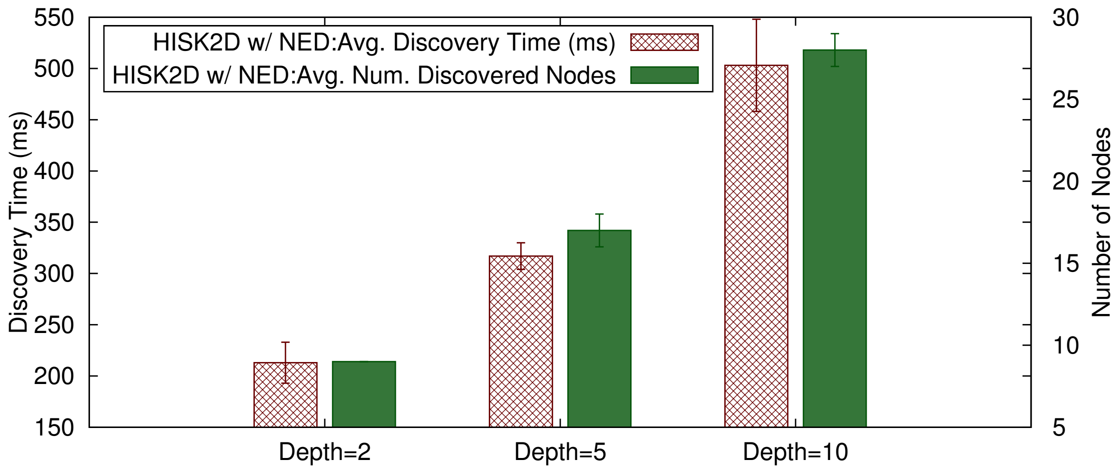

Figure 3.55: Average time to discover and number of discovered devices with background traffic high

HISK2D with NED spends 500 (ms) to discovery approximately 30 devices. The factor that most influences the discovery time is explained according to the cooperation between devices to depths reuse functionality (Section 3.3). For example, the devices do not search over the whole network until reaching the maximum depth, and just some of them update the management information gathered from the others, without making any further demands. This contributes to an efficient devices discovery process, according to the acquired knowledge of the network. Moreover, in order to maintain the information always updated, the partial views of each node consider the most recent information exchanged as well.

3.7 Concluding Remarks

Section 3.1 presented the proposed concept for mobile communication infrastructure integrating all mechanisms proposed and evaluated in this Thesis. Initial communication, exchange of management information, network (re) organization and data dissemination were the main targeted research topics in which most of the proposed mechanisms were devised and analyzed.

Sections 3.2, 3.3, 3.4 and 3.5 presented several important communication mechanisms that aim at providing dynamic cooperation over the top of a mobile communication infrastructure.

Regarding the initial communication presented in the Section 3.2, traditional mechanisms mostly require excessive overhead to collect, synchronize and disseminate management information for the direct connected nodes. In response to this, we proposed mechanisms for bootstrapping and discovery to ensure the proper base, following the Hide and Seek concept, where the nodes change their role dynamically according to events in the network, and with dynamic probing intervals according to the number of Seekers entering or leaving the network. The results presented the benefits provided by discovery through Hide and Seek mechanism in terms of control overhead for node discovery, convergence time and exchange of initial information between the surrounding 1-hop nodes.

However, 1-hop is not sufficient to provide a communication base for distributed management approaches. Thus, the exchange of information beyond the limit of 1-hop nodes was addressed through the NED function, which was proposed and integrated in the Hide and Seek mechanism. The results of this integration showed that NED provides better scalability, robustness and improves efficiency to exchange management information compared to the other approaches. At this point, we provide mechanisms for initial communication and exchange of information beyond the limit of directly connected nodes.

Autonomic and distributed management requires the cooperation between the network nodes or a group of nodes, and the periodical exchange of network information between them. Working in the MAC layer permits a more efficient and scalable information-oriented service at upper layers. Section 3.4 addressed the use of social-based metrics for node and network self-organization into quality communities, with the purpose of improving their message exchange in support of management decentralization in highly dynamic wireless networks. Loop free operation and partitioning problems were also addressed through distributed algorithms, avoiding permanent loops and misleading information caused by partitioning problems in the network. The obtained results show that the social-based metrics increase the stability of the links and decrease the required number of messages exchanged between nodes when compared to the baselines association. Moreover, a social network factor was integrated in the social-based metrics enforcing an enhanced version of the social-based metrics which is denoted by the name DtD. DtD uses GA as a solution to optimize the technical input weights, in order to maximize the dissemination of multimedia streaming, and at the same time minimize the distance to the information source nodes as well as spending a lower average load ratio in highly mobile environments, as presented in the Section 3.5.

Finally, the Section 3.6 presented the proof-of-concept from a practical point of view in order to prove the feasibility of the communication framework proposed for distributed network management. The HISK2D with the NED function, as part of the initial communication (Sections 3.2) and exchange of management information (Section 3.3) stages were the mechanisms used in the testbed. The performance analysis show that the communication framework has real conditions to unify distributed management over different network topologies and technologies, proving to be a lightweight easy-respond framework in terms of time to detect/recover and update faults, CPU overhead and average time/number of devices discovered, according to the acquired knowledge of the network.

Chapter 4

Conclusion and Future Research

In this Section, the main conclusions of this Thesis are presented, followed by the open issues for future research.

4.1 General Conclusions

Communication between nodes for self and distributed management covers a vast research area which has many open issues. Several innovative ideas covering different stages of communication between nodes have been proposed to overcome limitations pointed out by the literature. We investigated techniques, mechanisms and solutions to understand and identify the main challenges and requirements in order to propose a lightweight communication infrastructure for communication between nodes. Those challenges led to the definition and evaluation of several new mechanisms with particular emphasis on bootstrapping and hide and seek discovery, fault detection, exchange of management information through NED, social network associations for network (re) organization, loop and partitioning detection and enhancements for data dissemination. Moreover, the main goal of distributed management is to achieve scalable, robust management systems with low complexity for large-scale, dynamic network environments. In order to achieve these goals a number of functional requirements are presented in the state-of-the-art (Section 2.3.2), where we identified and classified the most important ones, such as situation awareness, level of integration, adaptability, extensibility, scalability and comprehensiveness of management information. Thus, we followed these main requirements in order to propose and develop a communication framework for distributed network management as presented in this Thesis. Along this Thesis, we also followed a concrete line of research that involved the definition and evaluation of a set of mechanisms in order to meet the research objectives described in the Chapter 1.2 of this Thesis, presented in more detail in the following annexes.

4.2 Conclusions of Research Objectives

The research objectives addressed by this Thesis were identified as:

4.2.1 Research Objective I

In which way must the nodes exchange initial information to reduce communication costs?

An important requirement over distributed network management approaches is the support for initial communication between nodes. In response to this research objective, HIde and SeeK Directional Decision (HISK2D) mechanism (more detail in the Annex A) was proposed which considered bootstrapping and discovery as two essential mechanisms to ensure proper initial communication between nodes. Through the concept of Hide and Seek, the entities are able to change their role dynamically according to events in the network, with dynamic probing intervals according to the number of Seekers entering or leaving the network. The important characteristics detected and stressed in this research work by the analysis of the proposed mechanisms are: (i) nodes can change dynamically their roles (Hider or Seeker) according to the network situation and events, reducing unnecessary signaling messages; (ii) dynamic refresh intervals help to reduce the impact of the overall network overhead; (iii) events and triggers are executed in accordance with the type of the node contacted, optimizing the communication between nodes. The results showed that the proposed mechanism has a lower impact on network overhead and lower convergence time for node discovery and synchronization of the gathered knowledge.

4.2.2 Research Objective II

How to ensure the exchange of partial information between non directly connected nodes?

Node cooperation is a key part of distributed management process, requiring periodic exchange of management information between directly and non directly connected nodes. In response to this research objective, HISK2D uses a Neighbors Eyesight Direction (NED) function (greater detail in the Annex A and B), which aims to narrow the directions, exchanging the partial information of the network until the last hop device (depth limit). Despite the assessments made in wired environments through Grid topologies ranging from 16 up to 81 virtual nodes, the NED demonstrated to exchange management information without exerting extra cost in the network. The NED behavior through numerical simulation in two different wireless scenarios, random and realistic device placement, was analyzed. The NED proved to be scalable for the evaluated scenarios, and also enforces the devices to search the best neighbor direction. It contributes to improve resource aggregations, reducing the dissemination cost and the device's resources consumption, in the exchange of management information for dense wireless networks.

4.2.3 Research Objective III

How to maintain, update and disseminate management and data information in highly dynamic environments?

In response to this objective, self-organizing decentralized wireless management through social-based metrics is an alternative to maintain the management information constantly updated (greater detail in the Annex C) under unstable network conditions. In practical terms, the given benefits are achieved without modifying the operating mode of the protocol where they were implemented (IEEE 802.11 MAC layer), but rather complementing it with new Information Elements that were included into the standard management frames. This need came due to the fact that the vast majority of wireless communication devices use this standard as the basis for communication. The proposed metrics have a positive impact in a distributed management approach, reducing the disconnection between nodes and updating the information exchange between them. The obtained results show that the proposed metrics increase the stability of the links and decrease the required number of messages exchanged between nodes when compared to the baselines (without social metrics) association. As a concrete example, in the case of dissemination of multimedia streaming, the challenge increases due to the data content size and volume and the partitioning problem, causing the network to be splitted into several smaller isolated end-to-end groups or subnets. The mechanisms for dissemination need to automatically ensure that all multimedia content will be disseminated for the largest number of devices possible (e.g. until the last-hop devices), with low end-to-end-delay and maximizing success delivery rates. The results of simulations, conducted through two case studies, have shown that our proposed solution tends to be more efficient in terms of: reception quality of multimedia streams, distance to stream source, and average load ratio, according to the baselines analyzed (greater detail in the Annex D).

4.2.4 Research Objective IV

How to provide proper management information to autonomic decision systems?

Network decisions require minimum valid amount of information in order to take decisions to improve network performance. In response to this research objective, the communication framework provides proper management information to the decision systems, due to the possibility to supply and control the amount of information to be exchanged in the network, by enabling cooperation between multi-network topologies, devices and technologies. This process allows, based on local knowledge to a specified depth, the possibility of limiting the amount of knowledge to analyze and obtain a solution similar to the decision made by a centralized system that would use all available information about the network.

4.2.5 Research Objective V

How to integrate the objectives I, II, III and IV into a communication infrastructure to support self and autonomic management over heterogeneous environments?

From the conceptual point of view, the communication infrastructure integrating the mechanisms proposed in this research work is presented in the Figure 3.1 from Chapter 3. Herewith, the interaction between the mechanisms for initial communication, exchange of information and network (re) organization and data dissemination are presented. Moreover, from a practical point of view, the mechanisms for discovery and exchange of management information that compose the communication framework were assessed in a testbed which involves different standards (IEEE 802.3, IEEE 802.11and IEEE 802.16e) and network technologies, such as wired virtual grid, wireless ad-hoc gateways, mobile access devices and WIMAX as long range technology (Section 3.6 from Chapter 3). This evaluation shows that is possible to unify distributed network management functionalities into the managed devices on top of distributed multi-network technologies scenarios. Moreover, the proposed communication framework overcomes the challenge of maintaining the distributed management information for both wired and wireless environment reaching the last-hop device without compromising the network performance.

4.3 Future Research

Within the scope of the conducted research work, there are numerous possible paths for further research that can directly complement the communication infrastructure proposed in this Thesis, along with its control and management mechanisms. Certainly, there are several aspects that have room for improvement in the presented research, and these aspects can create opportunities for a more advanced study of several concepts addressed in this Thesis. Herewith we presented some open issues for future research.

To comply with this open vision, there are many scenarios that open the possibility to assess the communication infrastructure proposed in in Thesis, such as, vehicular networks, wireless mesh and critical situations, disasters networks, etc. A field that is gaining attention is the ad-hoc vehicular networks (VANETs), where vehicles can cooperate through vehicle-to-vehicle communications, enabling interesting applications and services through multimedia video streaming (e.g. warning car drivers about any obstacle ahead through real time streaming video between the vehicles). In terms of transportation scenarios, such as, trains, subways and buses, there is a high possibility to provide entertainment to people through sharing multimedia streaming between passengers (e.g. watching movies, sharing photos, videos or audio files) before arriving at the destination point.

As further aspects to develop, there are possibilities of extending the sets of decision making, as in response to a malfunction (self-healing), considering the existence of partial information.

One possible solution to further decrease the signaling overhead is the introduction of Bloom filters [BMB02, LH06], with the main idea of creating filters that can be encapsulated in the data field of the signaling messages. Thus, instead of sending partial information, those nodes will send only a set of meta-filters through which the information may be located in the network using a query-and-response system, decreasing even more the overhead impact in the network. Additionally, all mechanisms engineered in this Thesis addressed common problems and challenges for many other areas of research; for example, Software Defined Networks (SDNs), dynamic routing, delay tolerant networks, pre-routing, seamless mobility and multi-homing. The SDN [Fun12] architecture proposes changes in the network through a manageable, cost-effective, and adaptable architecture, separating the data and control planes. This architecture decouples the network control and forwards functions enabling the network control to become directly programmable. Even though SDNs use centralized management controllers, the network end-points still have room for innovation. Thus, our mechanisms for discovery and network (re) organization are able to provide initial communication between switches at layer-2, and also support the lightweight information exchange in case of wireless scenarios [Fun13].

As part of the future work, we further envision the possibility of incorporating socio-technical user behavior patterns through the proposed social-based metrics into network infrastructure, enabling a virtuous cycle of innovation and creating greater synergies between social networks and mobile networks. Finally, the work of this Thesis should not only contribute to a compiled set of results and contributions, but should also stimulate the integration of novel paradigms to help this work evolve through other domains. It is then important to apply the knowledge gained throughout this Thesis to define other lines of research that could extend the range of concepts explored under the vision of the proposed mechanisms for communication between nodes.

Bibliography

[Ab10] E. Ab. D4.3 - In-Network Management Design. Technical report, FP7-EU-ICT, 2010. Available at `http://www.4ward-project.eu/index.php?s=file_download&id=77`.

[ABCA06] P. Aranda, A.-M. Biraghi, J. Carapinha, and M. Achemlal. D-2.1 technical requirements. Technical report, 2006. Available at `http://www.4ward-project.eu/index.php?s=file_download&id=34`.

[AD99] E. Aboelela and C. Douligeris. Fuzzy temporal reasoning model for event correlation in network management. In *Local Computer Networks, 1999. LCN '99. Conference on*, pages 150–159, 1999.

[ADM+09] Alberto Gonzalez Prieto, D. Dudkowski, C. Meirosu, C. Mingardi, G. Nunzi, M. Brunner, and R. Stadler. Decentralized in-network management for the future internet. *2009 IEEE International Conference on Communications Workshops*, pages 1–5, June 2009. Available at `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5207964`.

[Aea96] E. Amir and et al. Group communication as an infrastructure for distributed system management. In *Services in Distributed and Networked Environments, 1996., Proceedings of Third International Workshop on*, pages 84 –91, jun 1996.

[AIH97] J. Anderson, M. Ilyas, and S. Hsu. Distributed network management in an internet environment. In *Global Telecommunications Conference, 1997. GLOBECOM '97., IEEE*, volume 1, pages 180–184 vol.1, 1997.

[AKET09] G. Athanasiou, T. Korakis, O. Ercetin, and L. Tassiulas. A cross-layer framework for association control in wireless mesh networks. *Mobile Computing, IEEE Transactions on*, 8(1):65 –80, jan. 2009.

[ANA06] P. ANA. Ana: Autonomic network architecture, 2006. Available at `http://www.ana-project.org/`.

[Aut08] AutoI. Seventh programme eu-ist autonomic internet. *AUTO-I*, 2008. Available at `http://ist-autoi.eu/autoi/`.

[AutaN] Autonomic. Autonomic manifesto. *IBM*, NaN. Available at `www.research.ibm.com/autonomic/manifesto`.

[Awa13] Awareness. Self-awareness in autonomic systems, 2010-2013. Available at `http://www.aware-project.eu/about/`.

[AYB10] A. Abbasi, M. Younis, and U. Baroudi. Restoring connectivity in wireless sensor-actor networks with minimal topology changes. In *Communications (ICC), 2010 IEEE International Conference on*, pages 1–5, 2010.

[bat08] Batman(better approach to mobile ad-hoc networking, 2008. Available at `http://tools.ietf.org/html/draft-wunderlich-openmesh-manet-routing-00`.

[BDMN09] M. Brunner, D. Dudkowski, C. Mingardi, and G. Nunzi. Probabilistic decentralized network management. In *Integrated Network Management, 2009. IM '09. IFIP/IEEE International Symposium on*, pages 25–32, 2009.

[Bea00] A. M. Barotto and et al. Distributed network management using snmp, java, www and corba. *Journal of Network and Systems Management*, 8(4):483–497, 2000. Available at `http://dx.doi.org/10.1023/A:1026430416255`.

[BEM07] S. A. Borbash, A. Ephremides, and M. J. McGlynn. An asynchronous neighbor discovery algorithm for wireless sensor networks. *Ad Hoc Netw.*, 5(7):998–1016, September 2007. Available at `http://dx.doi.org/10.1016/j.adhoc.2006.04.006`.

[Bio09] Bionets Project. Biologically inspired network and services, 2009. Available at `http://www.bionets.eu/`.

[BJM+05] O. Babaoglu, M. Jelasity, A. Montresor, C. Fetzer, S. Leonardi, A. van Moorsel, and M. van Steen. *Self-Star Properties in Complex Information Systems.* Springer, 2005.

[BM09] M. Bratislav and M. Miroslaw. Npart - node placement algorithm for realistic topologies in wireless multihop network simulation. In *SIMUTools*, 2009.

[BMB02] A. Broder, M. Mitzenmacher, and A. Broder. Network Applications of Bloom Filters: A Survey. In *Internet Mathematics*, volume 1, pages 636–646, 2002.

[Bon] BonnMotion. A mobility scenario generation and analysis tool, copyright 2002-2011 university of bonn. Available at `http://net.cs.uni-bonn.de/wg/cs/applications/bonnmotion/`.

[BPM+10] A. Bashar, G. Parr, S. McClean, B. Scotney, M. Subramanian, S. Chaudhari, and T. Gonsalves. Employing bayesian belief networks for energy efficient network management. In *Communications (NCC), 2010 National Conference on*, pages 1–5, 2010.

[BPW98] A. Bieszczad, B. Pagurek, and T. White. Mobile agents for network management. *IEEE Communications Surveys, Vol. 1 No 1*, Fourth Quarter 1998. Available at `http://citeseer.ist.psu.edu/bieszczad98mobile.html`.

[BS09] N. Bhalaji and A. Shanmugam. Association between nodes to combat blackhole attack in dsr based manet. In *Wireless and Optical Communications Networks, 2009. WOCN '09. IFIP International Conference on*, pages 1 –5, april 2009.

[BS13] M. Barshan and M. Shojaei. Management challenges and solutions in next-generation networks (ngn). In V. Das and Y. Chaba, editors, *Mobile Communication and Power Engineering*, volume 296 of *Communications in Computer and Information Science*, pages 549–555. Springer Berlin Heidelberg, 2013.

[BSF06] R. Badonnel, R. State, and O. Festor. Probabilistic management of ad-hoc networks. In *Network Operations and Management Symposium, 2006. NOMS 2006. 10th IEEE/IFIP*, pages 339–350, 2006.

[CBP00] R. R. Chpudhury, S. Bandyopadhyay, and K. Paul. A distributed mechanism for topology discovery in ad hoc wireless networks using mobile agents. In *Proc. MobiHOC Mobile and Ad Hoc Networking and Computing 2000 First Annual Workshop on*, pages 145–146, 11 Aug. 2000.

[Cea90] J. Case and et al. *IETF RFC 1157 - Simple Network Management Protocol (SNMP)*, May 1990. Available at `http://www.ietf.org/rfc/rfc1157.txt`.

[Cea02] T. M. Chen and et al. A model and evaluation of distributed network management approaches. *IEEE Journal on Selected Areas in Communications*, 20(4):8, May 2002.

[CHLS07] C. Chi, D. Huang, D. Lee, and X. Sun. Lazy flooding: A new technique for information dissemination in distributed network systems. *IEEE/ACM Trans. Netw.*, 15(1):80–92, Feb. 2007.

[CJL03] T. Clausen, P. Jacquet, and A. Laouiti. *Optimized Link State Routing Protocol (OLSR)*. IETF RFC 3626, October 2003. Available at `http://www.ietf.org/rfc/rfc3626.txt`.

[CJSR08] G. W. Cox, W. L. Johnson, J. Strassner, and D. Raymer. Bootstrapping device state in managed systems. In *Proc. IEEE Network Operations and Management Symposium NOMS 2008*, pages 767–770, 7–11 April 2008.

[CK08] J. Camp and E. Knightly. The ieee 802.11s extended service set mesh networking standard. *Communications Magazine, IEEE*, 46(8):120 –126, august 2008.

[Cle07] A. Clemm. *Network Management Fundamentals: A Guide to understanding how network management technology really works.* Cisco Press, 2007. Available at `http://my.safaribooksonline.com/1587201372?portal=ciscopress`.

[CMAB13] G. Chasparis, M. Maggio, K.-E. Arzen, and E. Bini. Distributed management of cpu resources for time-sensitive applications. In *American Control Conference (ACC), 2013*, pages 5305–5312, 2013.

[CMLD10] Y. Cui, T. Ma, J. Liu, and S. Das. Load-balanced ap association in multi-hop wireless mesh networks. *The Journal of Supercomputing*, pages 1–27, December 2010. Available at `http://dx.doi.org/10.1007/s11227-010-0519-7`.

[CMS08] C. Chen, J. Ma, and J. Salomaa. Simulation study of cluster based data dissemination for wireless sensor networks with mobile sinks. In *Advanced Communication Technology, 2008. ICACT 2008. 10th International Conference on*, volume 1, pages 231–236, 2008.

[CMT08] D. Charilas, O. Markaki, and E. Tragos. A theoretical scheme for applying game theory and network selection mechanisms in access admission control. In *Wireless Pervasive Computing, 2008. ISWPC 2008. 3rd International Symposium on*, pages 303–307, 2008.

[Coo11] O. Cooporation. Fing discovery protocol, Accessed in 2011. Available at http://www.over-look.com/site/index.php.

[CS03] S. Calo and M. Sloman. Guest editorial: Policy-based management of networks and services. *Journal of Network and Systems Management*, 11(3):249–252, 2003. Available at http://dx.doi.org/10.1023/A:1025728221539.

[CSGW10] P. Chen, J. Su, W. Guo, and J. Wei. The implementation of a new topology discovery algorithm in mobile ad hoc network. In *Multimedia Technology (ICMT), 2010 International Conference on*, pages 1–4, 2010.

[CWBM+13] R. Chaparadza, M. Wodczak, T. Ben Meriem, P. De Lutiis, N. Tcholtchev, and L. Ciavaglia. Standardization of resilience amp; survivability, and autonomic fault-management, in evolving and future networks: An ongoing initiative recently launched in etsi. In *Design of Reliable Communication Networks (DRCN), 2013 9th International Conference on the*, pages 331–341, 2013.

[Dar09] W. Dargie. *Context-Aware Computing and Self-Managing Systems*, volume 1. CRC press, 2009. Available at http://books.google.pt/books?id=Y0JtqV5ODzQC&lpg=PP1hl=pt-PT&pg=PP1#v=onepage&q&f=false.

[DAS09] H. Derbel, N. Agoulmine, and M. Salaün. Anema: Autonomic network management architecture to support self-configuration and self-optimization in ip networks. *Computer Networks*, 53(3):418–430, 2009. Available at http://www.sciencedirect.com/science/article/B6VRG-4TW14YJ-3/2/41147806c839b89928697fc9b724d880.

[DBN+09] D. Dudkowski, M. Brunner, G. Nunzi, C. Mingardi, C. Foley, M. De Leon, C. Meirosu, and S. Engberg. Architectural principles and elements of in-network management. In *Integrated Network Management, 2009. IM '09. IFIP/IEEE International Symposium on*, pages 529–536, 2009.

[DBS+07] S. Davy, K. Barrett, M. Serrano, J. Strassner, B. Jennings, and S. Van der Meer. Policy interactions and management of traffic engineering services based on ontologies. In *Network Operations and Management Symposium, 2007. LANOMS 2007. Latin American*, pages 95–105, 2007.

[DC08] P. Dutta and D. Culler. Practical asynchronous neighbor discovery and rendezvous for mobile sensing applications. In *Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems*, SenSys '08, pages 71–84, New York, NY, USA, 2008. ACM. Available at http://doi.acm.org/10.1145/1460412.1460420.

[DDF+06] S. Dobson, S. Denazis, A. Fern #225, ndez, D. Ga #239, ti, E. Gelenbe, F. Massacci, P. Nixon, F. Saffre, N. Schmidt, and F. Zambonelli. A survey of autonomic communications. *ACM Trans. Auton. Adapt. Syst.*, 1(2):223–259, 2006.

[DDL+02] N. Damianou, N. Dulay, E. Lupu, M. Sloman, and T. Tonouchi. Tools for domain-based policy management of distributed systems. In *Network Operations and Management Symposium, 2002. NOMS 2002. 2002 IEEE/IFIP*, pages 203–217, 2002.

[DFKS07] V. Drabkin, R. Friedman, G. Kliot, and M. Segal. Rapid: Reliable probabilistic dissemination in wireless ad-hoc networks. In *Proc. 26th IEEE International Symposium on Reliable Distributed Systems SRDS 2007*, pages 13–22, 10–12 Oct. 2007.

[DG09] D.Dudkowski and G.Nunzi. D4.2 in-network management concept. Technical report, 4-WARD, 17 February 2009. Available at `http://www.4ward-project.eu/index.php?s=file_download&id=77`.

[DH07] E. M. Daly and M. Haahr. Social network analysis for routing in disconnected delay-tolerant manets. In *Proceedings of the 8th ACM international symposium on Mobile ad hoc networking and computing*, MobiHoc '07, pages 32–40, New York, NY, USA, 2007. ACM. Available at `http://doi.acm.org/10.1145/1288107.1288113`.

[Din10] J. Ding. *Advances in Network Management.* CRC Press, 2010. Available at `http://books.google.com.br/books?id=JOSUimXOoGMC&lpg=PP1&hl=pt-PT&pg=PP1#v=onepage&q&f=false`.

[DKBC05] J. Ding, B. Krämer, Y. Bai, and H. Chen. Backward inference in bayesian networks for distributed systems management. *Journal of Network and Systems Management*, 13(4):409–427, 2005. Available at `http://dx.doi.org/10.1007/s10922-005-9003-8`.

[DL05] G. Dong and J. Liebeherr. The spanning tree protocol. Technical report, MNG Group, 2005. Available at `http://www.comm.utoronto.ca/hypercast/material/SPT_Protocol_03-20-05.pdf`.

[DLW93] R. H. Deng, A. A. Lazar, and W. Wang. A probabilistic approach to fault diagnosis in linear lightwave networks. In *Proceedings of the IFIP TC6/WG6.6 Third International Symposium on Integrated Network Management with participation of the IEEE Communications Society CNOM and with support from the Institute for Educational Services*, pages 697–708. North-Holland Publishing Co., 1993.

[DO03] D. Doval and D. O'Mahony. Overlay networks: A scalable alternative for p2p. *IEEE Internet Comput.*, 7(4):79–82, July–Aug. 2003.

[DXT09] T. Dinh, Y. Xuan, and M. Thai. Towards social-aware routing in dynamic communication networks. In *Performance Computing and Communications Conference (IPCCC), 2009 IEEE 28th International*, pages 161 –168, dec. 2009.

[EL02] J. Eidson and K. Lee. Ieee 1588 standard for a precision clock synchronization protocol for networked measurement and control systems. In *Sensors for Industry Conference, 2002. 2nd ISA/IEEE*, pages 98–105, 2002.

[FBP+08] C. Foley, S. Balasubramaniam, E. Power, M. P. Leon, D. Botvich, D. Dudkowski, G. Nunzi, and C. Mingardi. A framework for in-network management in heterogeneous future communication networks. In *Modelling Autonomic Communications Environments*, pages 14–25. Springer-Verlag, 2008.

[FDT13] J. Famaey and F. De Turck. Federated and autonomic management of multimedia services. In *Integrated Network Management (IM 2013), 2013 IFIP/IEEE International Symposium on*, pages 927–933, 2013.

[Fea06] C. Farnham and et al. A market-based optimization approach to sensor and resource management. *Proc. SPIE, Vol. 6229, 62290I (2006); doi:10.1117/12.665153*, 17 April 2006.

[FECCP13] M. Fiore, C. Ettore Casetti, C. Chiasserini, and P. Papadimitratos. Discovery and verification of neighbor positions in mobile ad hoc networks. *Mobile Computing, IEEE Transactions on*, 12(2):289–303, 2013. 10.1109/TMC.2011.258.

[FKALG+06] R. Farha, M. S. Kom, Alberto-Leon-Garcia, J. W.-K. Hong, and Yu Cheng. A generic architecture for autonomic service and network management. *Computer Communications*, 29(18):3691–3709, November 2006.

[FNJ10] Y. Fayyaz, M. Nasim, and M. Javed. Maximal weight topology discovery in ad hoc wireless sensor networks. In *Computer and Information Technology (CIT), 2010 IEEE 10th International Conference on*, pages 715–722, 2010.

[Fun12] O. N. Fundation. Software-defined networking:the new norm for networks. *White Paper*, April 2012. Available at `https://www.opennetworking.org/images/stories/downloads/sdn-resources/white-papers/wp-sdn-newnorm.pdf`.

[Fun13] O. N. Fundation. Openflow-enabled mobile and wireless networks. *White Paper*, September 30, 2013. Avaiable at `https://www.opennetworking.org/images/stories/downloads/sdn-resources/solution-briefs/sb-wireless-mobile.pdf`.

[FV11] F. Fabbri and R. Verdone. A sociability-based routing scheme for delay-tolerant networks. *EURASIP J. Wirel. Commun. Netw.*, 2011:1:1–1:13, January 2011. Available at `http://dx.doi.org/10.1155/2011/251408`.

[GCG+13] L. Guardalben, T. Condeixa, T. Gomes, P. Salvador, and S. Sargento. On the analysis of dissemination management information through an eyesight perspective. In *5th IEEE International Workshop on Management of Emerging Networks and Services (IEEE Globecom MENS Workshop)*, December 2013.

[GG08] C. GauthierDickey and C. Grothoff. Bootstrapping of peer-to-peer networks. In *Proceedings of the 2008 International Symposium on Applications and the Internet*, SAINT '08, pages 205–208, Washington, DC, USA, 2008. IEEE Computer Society. Available at `http://dx.doi.org/10.1109/SAINT.2008.15`.

[GGP+11] L. Guardalben, T. Gomes, A. Pinho, P. Salvador, and S. Sargento. Nodes discovery in the in-network management communication framework. In *MONAMI*, pages 145–157, 2011.

[GGS+13] L. Guardalben, T. Gomes, R. Sousa, P. Salvador, and S. Sargento. Lightweight discovery and exchange of network information in distributed network management. Ghent, Belgium, Integrated Management IM May 2013.

[GGSS] L. Guardalben, T. Gomes, P. Salvador, and S. Sargento. Multimedia streaming dissemination through cooperative device-to-device behaviors in mobile networks. In *Submitted to IEEE JSAC Special Issue on Device-to-Device Communications in Cellular Networks.*

[GGSS12a] L. Guardalben, T. Gomes, P. Salvador, and S. Sargento. Improving mac layer association through social-based metrics in mobile networks. *IEEE Communications Magazine*, 50(6):91–98, 2012.

[GGSS12b] L. Guardalben, T. Gomes, P. Salvador, and S. Sargento. Self-organizing decentralized wireless management through social-based metrics. In *Network Operations and Management Symposium (NOMS), 2012 IEEE*, pages 1199–1202, 2012.

[GGSS13a] T. Gomes, L. Guardalben, P. Salvador, and S. Sargento. A genetic algorithm approach to improve network nodes association. In *Globecom 2013 - Communications QoS, Reliability and Modelling Symposium*, jun 2013.

[GGSS13b] T. Gomes, L. Guardalben, P. Salvador, and S. Sargento. A spanning tree protocol over mobile wireless ad hoc networks. In *IEEE International Conference on Communications 2013: IEEE ICC'13 - 3rd IEEE International Workshop on Smart Communication Protocols and Algorithms (SCPA 2013) (ICC'13 - IEEE ICC'13 - Workshop SCPA)*, pages 1109–1114, Budapest, Hungary, June 2013.

[GKP+09] V. Gazis, A. Kousaridas, C. Polychronopoulos, T. Raptis, and N. Alonistioti. Self-management capacities in future internet wireless systems. In *Future Computing, Service Computation, Cognitive, Adaptive, Content, Patterns, 2009. COMPUTATIONWORLD '09. Computation World:*, pages 9–14, 2009.

[GKW+12] S. Gogouvitis, K. Konstanteli, S. Waldschmidt, G. Kousiouris, G. Katsaros, A. Menychtas, D. Kyriazis, and T. Varvarigou. Workflow management for soft real-time interactive applications in virtualized environments. *Future Generation Computer Systems*, 28(1):193 – 209, 2012.

[GO07] Z. Genc and O. Ozkasap. Eramobile: Epidemic-based reliable and adaptive multicast for manets. In *Wireless Communications and Networking Conference, 2007.WCNC 2007. IEEE*, pages 4395–4400, 2007.

[Gol89] D. E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1st edition, 1989.

[Gom12] T. Gomes. Comunicação entre os elementos da rede numa gestão autonómica. Master's thesis, DETI/Universidade de Aveiro, 2012. Available at `https://ria.ua.pt/handle/10773/7378`.

[Gro10] GroundWork. Groundwork open source it monitoring and network management software, http://www.groundworkopensource.com/, 2010.

[GRSC98] S. Ghosh, Q. Razouqi, H. J. Schumacher, and A. Celmins. A survey of recent advances in fuzzy logic in telecommunications networks and new challenges. *Fuzzy Systems, IEEE Transactions on*, 6(3):443–447, 1998.

[Gup06] A. Gupta. Network management: Current trends and future perspectives. *Journal of Network and Systems Management*, 14(4):483–491, December 2006.

[GVMS10] L. Guardalben, P. S. Vitor Mirones, and S. Sargento. A cooperative hide and seek discovery over in network management. *2nd IFIP/IEEE International Workshop on Management of the Future Internet (ManFi 2010), IEEE/IFIP Network Operations and Management Symposium*, January 2010.

[GZJL09] L. Gu, C. Zhang, Y. Ji, and L. Li. An o(1) lookup and decentralized bootstrapping peer to peer sip system. In *Consumer Communications and Networking Conference, 2009. CCNC 2009. 6th IEEE*, pages 1–2, 2009.

[HAG06] HAGGLE. Haggle (an innovative paradigm for autonomic opportunistic communication). *HAGGLE*, 2006. Available at `http://www.haggleproject.org/`.

[HCY11] P. Hui, J. Crowcroft, and E. Yoneki. Bubble rap: Social-based forwarding in delay-tolerant networks. *Mobile Computing, IEEE Transactions on*, 10(11):1576 –1589, nov. 2011.

[HG98] J. Hung and J. Gutierrez. The application of web-based technologies on integrated network management. In *Global Telecommunications Conference, 1998. GLOBECOM 1998. The Bridge to Global Integration. IEEE*, volume 2, pages 1094–1098 vol.2, 1998.

[HMZ08] X. Huang, Y. Ma, and J. Zhang. Study of the active network management system model based on agent. In *Wireless Communications, Networking and Mobile Computing, 2008. WiCOM '08. 4th International Conference on*, pages 1–4, 2008.

[HSB10] S. Hariharan, N. Shroff, and S. Bagchi. Secure neighbor discovery through overhearing in static multihop wireless networks. In *Wireless Mesh Networks (WIMESH 2010), 2010 Fifth IEEE Workshop on*, pages 1–6, 2010.

[HYY+13] Q. He, J. Yan, Y. Yang, R. Kowalczyk, and H. Jin. A decentralized service discovery approach on peer-to-peer networks. *Services Computing, IEEE Transactions on*, 6(1):64–75, 2013.

[IBM11] IBM. Ibm tivoli monitoring software, 2011. Available at `http://www-01.ibm.com/software/tivoli/products/monitor/`.

[ICH08] P. Israt, N. Chakma, and M. M. A. Hashem. A fuzzy logic-based adaptive handoff management protocol for next-generation wireless systems. In *Computer and Information Technology, 2008. ICCIT 2008. 11th International Conference on*, pages 288–293, 2008.

[IEE07] IEEE std. 802.11-2007, part 11: wireless LAN medium access control (MAC) and physical layer (PHY) specifications, June 2007. Available at `http://standards.ieee.org/findstds/standard/802.11-2007.html`.

[ift13] iftop: display bandwidth usage on an interface, Accessed 2013. Available at `http://www.ex-parrot.com/pdw/iftop/`.

[Inc13] C. Inc. Cisco visual networking index: Global mobile data traffic forecast 2012/2017. *White Paper*, February 6, 2013. Available at `http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-520862.pdf`.

[ipe13] NLANR/DAST : Iperf - the TCP/UDP bandwidth measurement tool, Accessed 2013. Available at `http://dast.nlanr.net/Projects/Iperf/`.

[ITU96] ITU-T. M.3010 principles for a telecommunications management network, 1996. Available at `http://www.itu.int/rec/T-REC-M.3010-200002-I/en`.

[ITU97] ITU-T. M.3400 tmn management functions, 1997. Available at `http://www.itu.int/rec/T-REC-M.3400-200002-I/en`.

[JGR08] A. Jabas, R. Garimella, and S. Ramachandram. Manet mining: Mining temporal association rules. In *Parallel and Distributed Processing with Applications, 2008. ISPA '08. International Symposium on*, pages 765 –770, dec. 2008.

[JLG10] Y. Jiang, C.-H. Lung, and N. Goel. A tree-based multiple-hop clustering protocol for wireless sensor networks. In J. Zheng, D. Simplot-Ryl, and V. Leung, editors, *Ad Hoc Networks*, volume 49 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pages 371–383. Springer Berlin Heidelberg, 2010.

[JMB05] M. Jelasity, A. Montresor, and O. Babaoglu. Gossip-based aggregation in large dynamic networks. *ACM Trans. Comput. Syst.*, 23(3):219–252, 2005.

[JMB09] M. Jelasity, A. Montresor, and O. Babaoglu. T-man: Gossip-based fast overlay topology construction. *Computer Networks*, 53(13):2321 – 2339, 2009. Gossiping in Distributed Systems.

[JP93] J. F. Jordaan and M. Paterok. Event correlation in heterogeneous networks using the osi management framework. In *Proceedings of the IFIP TC6/WG6.6 Third International Symposium on Integrated Network Management with participation of the IEEE Communications Society CNOM and with support from the Institute for Educational Services*, pages 683–695. North-Holland Publishing Co., 1993.

[JvB+07] B. Jennings, S. van der Meer, S. Balasubramaniam, D. Botvich, M. Foghlu, W. Donnelly, and J. Strassner. Towards autonomic management of communications networks. *Communications Magazine, IEEE*, 45(10):112–121, oct 2007.

[JZLZ13] M. Javed, K. Zen, H. B. Lenando, and H. Zen. Fast association process (fap) of beacon enabled for ieee 802.15.4 in strong mobility. In *Information Technology in Asia (CITA), 2013 8th International Conference on*, pages 1–8, 2013.

[KB97] M. Kahani and H. W. P. Beadle. Decentralised approaches for network management. *SIGCOMM Comput. Commun. Rev.*, 27(3):36–47, July 1997. Available at `http://doi.acm.org/10.1145/263932.263940`.

[KC03] J. O. Kephart and D. M. Chess. The vision of autonomic computing. In *IBM.*, volume 1, pages 41–50, 2003. Available at `http://users.soe.ucsc.edu/~griss/agent-papers/ieee-autonomic.pdf`.

[KCJ+08] S.-S. Kim, M.-J. Choi, H.-T. Ju, M. Ejiri, and J.-K. Hong. Towards management requirements of future internet. In M. Yan, C. Deokjai, and A. Shingo, editors, *Challenges for Next Generation Network Operations and Service Management*, volume 5297 of *Lecture Notes in Computer Science*, pages 156–166. Springer Berlin Heidelberg, 2008.

[KKS09] K. M. Konwar, D. Kowalski, and A. A. Shvartsman. Node discovery in networks. *Journal of Parallel and Distributed Computing*, 69(4):337 – 348, 2009. Available at `http://www.sciencedirect.com/science/article/B6WKJ-4V3HF9B-2/2/d4b1d66f80f6e3e62a69d91ed7da93ff`.

[KLS97] H. Ku, G. Luderer, and B. Subbiah. An intelligent mobile agent framework for distributed network management. In *Global Telecommunications Conference, 1997. GLOBECOM '97., IEEE*, volume 1, pages 160–164 vol.1, 1997.

[KNK06] G. Kasbekar, P. Nuggehalli, and J. Kuri. Online client-ap association in wlans. In *Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks, 2006 4th International Symposium on*, pages 1 – 8, april 2006.

[KPP+13] S. Kodithuwakku, L. Padmakumara, I. Premadasa, S. Rathnayaka, V. Nanayakkara, and S. Perera. Gajanindu: A distributed system management framework with user-defined management logic. In *Information Technology: New Generations (ITNG), 2013 Tenth International Conference on*, pages 83–88, 2013. 10.1109/ITNG.2013.125.

[KV97] G. P. Kumar and P. Venkataram. Artificial intelligence approaches to network management: recent advances and a survey. *Computer Communications*, 20(15):1313–1322, 1997. Available at `http://www.sciencedirect.com/science/article/B6TYP-3SNYTVW-15/2/30864683dbea0adefad6e04a279d9871`.

[KWSW07] M. Knoll, A. Wacker, G. Schiele, and T. Weis. Decentralized bootstrapping in pervasive applications. In *Proc. Fifth Annual IEEE International Conference on Pervasive Computing and Communications Workshops PerCom Workshops '07*, pages 589–592, 19–23 March 2007.

[KWSW08] M. Knoll, A. Wacker, G. Schiele, and T. Weis. Bootstrapping in peer-to-peer systems. In *Proc. 14th IEEE International Conference on Parallel and Distributed Systems ICPADS '08*, pages 271–278, 8–10 Dec. 2008.

[LCC+11] W. Li, Y. Cui, X. Cheng, M. Al-Rodhaan, and A. Al-Dhelaan. Achieving proportional fairness via ap power control in multi-rate wlans. *Wireless Communications, IEEE Transactions on*, PP(99):1 –9, 2011.

[LCT99] T. Luo, T. Confrey, and K. Trivedi. A reliable corba-based network management system. In *Communications, 1999. ICC '99. 1999 IEEE International Conference on*, volume 2, pages 1374–1387 vol.2, 1999.

[LDS03] A. Lindgren, A. Doria, and O. Schelén. Probabilistic routing in intermittently connected networks. *SIGMOBILE Mob. Comput. Commun. Rev.*, 7:19–20, July 2003. Available at `http://doi.acm.org/10.1145/961268.961272`.

[Lea95] K. H. Lee and et al. A multicast protocol for network management system. In *Networks, 1995. Theme: 'Electrotechnology 2000: Communications and Networks'. [in conjunction with the] International Conference on Information Engineering., Proceedings of IEEE Singapore International Conference on*, pages 364 –368, jul 1995.

[Lea08] M. Li and et al. Wbest: a bandwidth estimation tool for ieee 802.11 wireless networks. In *LCN, Montreal, Quebec, Canada*, 2008. Available at `ftp://ftp.cs.wpi.edu/pub/techreports/pdf/06-14.pdf`.

[LH06] F. Liu and G. J. Heijenk. In T. Braun, G. Carle, S. Fahmy, and Y. Koucheryavy, editors, *Proceedings 4th International Conference on Wired/Wireless Internet Communications, WWIC 2006, Bern, Switzerland*, volume 3970 of *Lecture Notes in Computer Science*, pages 13–25, Berlin, 2006. Springer-Verlag.

[LPR97] M. Leppinen, P. Pulkkinen, and A. Rautiainen. Java- and corba-based network management. *Computer*, 30(6):83–87, 1997. Available at `10.1109/2.587555`.

[LST13] R. Lazzarini, C. Stefanelli, and M. Tortonesi. Large-scale e-maintenance: A new frontier for management? In *Integrated Network Management (IM 2013), 2013 IFIP/IEEE International Symposium on*, pages 732–735, 2013.

[Lug08] G. Lugano. Mobile social networking in theory and practice. *First Monday*, 13(11), November 2008. Available at `http://firstmonday.org/htbin/cgiwrap/bin/ojs/index.php/fm/article/view/2232`.

[LWD13] L. Liu, Y. Wang, and J. Ding. Stochastic game model for information dissemination of emergency events in social network. In *Intelligence and Security Informatics (ISI), 2013 IEEE International Conference on*, pages 166–168, 2013.

[LZY07] Y. Liu, G. Zhu, and H. Yin. A practical hybrid mechanism for peer discovery. In *Proc. International Symposium on Intelligent Signal Processing and Communication Systems ISPACS 2007*, pages 706–709, Nov. 28 2007–Dec. 1 2007.

[Mat05] Matlab central. *Probabilistic Flooding Simulation*, 2005. Available at `http://www.mathworks.com/matlabcentral/fileexchange/7928-probabilistic-flooding-simulation`.

[Mat10] W. Mathworld. Acyclic digraph, 2010. http://mathworld.wolfram.com/AcyclicDigraph.html.

[MAT11] MATLAB. *version 7.13.0 (R2011b)*. The MathWorks Inc., 2011. Available at `http://www.mathworks.com/products/matlab/`.

[MB08] R. R. M. Buyya. An autonomic workflow management system for global grids. *Cluster Computing and the Grid, 2008. CCGRID '08. 8th IEEE International Symposium on*, 2008.

[MdX09] Z. M. Mao, J. V. der, and M. Xu Chen. Pacman: a platform for automated and controlled network operations and configuration management. *CoNEXT'09, December 1–4, 2009, Rome, Italy.*, 2009.

[Meg06] N. Meghanathan. An algorithm to determine the sequence of stable connected dominating sets in mobile ad hoc networks. In *Proc. International Conference on Internet and Web Applications and Services/Advanced International Conference on Telecommunications AICT-ICIW '06*, pages 32–32, 19–25 Feb. 2006.

[MFZH99] J.-P. Martin-Flatin, S. Znaty, and J.-P. Hubaux. A survey of distributed enterprise network andsystems management paradigms. *J. Netw. Syst. Manage.*, 7(1):9–26, 1999.

[MH08] A. Mawji and H. S. Hassanein. Bootstrapping p2p overlays in manets. In *Proc. IEEE Global Telecommunications Conference IEEE GLOBECOM 2008*, pages 1–5, Nov. 30 2008–Dec. 4 2008.

[MH10] A. Mawji and H. Hassanein. Implementation of bootstrapping for p2p overlays in manets. In *Communications (QBSC), 2010 25th Biennial Symposium on*, pages 14–17, 2010.

[MHZ08] Y. Ma, X. Huang, and J. Zhang. Design and analyze the model of the management system based on active network. In *Computing, Communication, Control, and Management, 2008. CCCM '08. ISECS International Colloquium on*, volume 3, pages 118–122, 2008.

[Mil05] B. Miller. The autonomic computing edge: Can you chop up autonomic computing?, jan 2005. Available at `http://www.ibm.com/developerworks/autonomic/library/ac-edge4/`.

[MLAV06] S. Milner, J. Llorca, A. Anibha, and U. Vishkin. A bootstrapping model for directional wireless networks. *IEEE Commun. Lett.*, 10(12):840–842, December 2006.

[MMDA10] A. Mtibaa, M. May, C. Diot, and M. Ammar. Peoplerank: Social opportunistic forwarding. In *INFOCOM, 2010 Proceedings IEEE*, pages 1 –5, march 2010.

[MMNK13] E. Mostajeran, R. Md Noor, and H. Keshavarz. A novel improved neighbor discovery method for an intelligent-aodv in mobile ad hoc networks. In *Information and Communication Technology (ICoICT), 2013 International Conference of*, pages 395–399, 2013. 10.1109/ICoICT.2013.6574608.

[MOKW06] H. A. Müller, L. OBrien, M. Klein, and B. Wood. Autonomic computing: Software architecture technology. Technical report, Carnegie Mellon University, april 2006. Available at `http://www.dtic.mil/dtic/tr/fulltext/u2/a448227.pdf`.

[Moy98] J. Moy. *OSPF Version 2*. IETF RFC 2328, April 1998. Available at `http://www.ietf.org/rfc/rfc2328.txt`.

[MPSJ02] R. Mundy, D. Partain, B. Stewart, and J. Case. Introduction and applicability statements for internet standard management framework. Technical report, IETF RFC 3410, 2002.

[MRW93] K. McCloghrie, M. Rose, and S. Waldbusser. Protocol operations for version 2 of the simple network management protocol. Technical report, IETF RFC 1448, 1993.

[MZ06] A. Manzalini and F. Zambonelli. Towards autonomic and situation-aware communication services: the cascadas vision. *Proceedings of the IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and Its Applications (DIS)*, pages 383–388, 2006.

[NBEK+08] N. Niebert, S. Baucke, I. El-Khayat, M. Johnsson, B. Ohlman, H. Abramowicz, K. Wuenstel, H. Woesner, J. Quittek, and L. Correia. The way 4ward to the creation of a future internet. In *Personal, Indoor and Mobile Radio Communications, 2008. PIMRC 2008. IEEE 19th International Symposium on*, pages 1–5, 2008.

[NDXT11] N. Nguyen, T. Dinh, Y. Xuan, and M. Thai. Adaptive algorithms for detecting community structure in dynamic social networks. In *INFOCOM, 2011 Proceedings IEEE*, pages 2282 –2290, april 2011.

[NND07] B. T. Nassu, T. Nanya, and E. P. Duarte. Topology discovery in dynamic and decentralized networks with mobile agents and swarm intelligence. In *Proc. Seventh International Conference on Intelligent Systems Design and Applications ISDA 2007*, pages 685–690, 20–24 Oct. 2007.

[NNPB10] S. Ning, M. Nabeel, F. Paci, and E. Bertino. A privacy-preserving approach to policy-based content dissemination. In *IEEE 26th International Conference on Data Engineering (ICDE)*, pages 944–955, 1-6 March 2010.

[NS-10] NS-3. Manual ns-3.9, 2010. Available at `http://www.nsnam.org/docs/release/3.9/manual.html`.

[Ost12] Ostinato. Packet/traffic generator and analyzer, 2012. Available at `http://code.google.com/p/ostinato/`.

[Pav99] J. Pavon. Building telecommunications management applications with corba. *Communications Surveys & Tutorials, IEEE*, 2(2):2–16, 1999.

[Pav07] G. Pavlou. On the evolution of management approaches, frameworks and protocols: A historical perspective. *J. Netw. Syst. Manage.*, 15(4):425–445, 2007.

[Pea99] P. Parnes and et al. A framework for management and control of distributed applications using agents and ip-multicast. In *INFOCOM '99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 3, pages 1445 –1452 vol.3, mar 1999.

[Pea02] E. Procopio and et al. A dependable snmp-based tool for distributed network management. In *Proceedings of the 2002 International Conference on Dependable Systems and Networks*, pages 279–284. IEEE Computer Society, 2002.

[Pet77] J. L. Peterson. Petri nets. *ACM Comput. Surv.*, 9(3):223–252, September 1977. http://doi.acm.org/10.1145/356698.356702.

[PHXQ10] H. Peng, P. Heng, L. Xiangdong, and Z. Qiusheng. Physical topology discovery based on spanning tree protocol. In *Computer Application and System Modeling (ICCASM), 2010 International Conference on*, volume 14, pages V14–308–V14–311, 2010.

[PM09] A. Pachon and J. Madrid. Application of an ant colony metaphor for network address management in manets. In *Communications, 2009. LATINCOM '09. IEEE Latin-American Conference on*, pages 1–6, 2009.

[PP09] G. Palla and Pollner. Social group dynamics in networks. In T. Gross and H. Sayama, editors, *Adaptive Networks*, volume 51 of *Understanding Complex Systems*, pages 11–38. Springer Berlin / Heidelberg, 2009.

[PR11] S. Preethi and B. Ramachandran. Energy efficient routing protocols for mobile adhoc networks. In *Emerging Trends in Networks and Computer Communications (ETNCC), 2011 International Conference on*, pages 136–141, 2011.

[RD01] A. Rowstron and P. Druschel, editors. *Pastry: Scalable, decentralized object location, i and routing for large-scale peer-to-peer systems.*, volume 2218. Lecture Notes in Computer Science, 2001.

[RGA+04] L. Raju, S. Ganu, B. Anepu, I. Seskar, and D. Raychaudhuri. Beacon assisted discovery protocol (bead) for self-organizing hierarchical ad-hoc networks. In *Global Telecommunications Conference, 2004. GLOBECOM '04. IEEE*, volume 3, pages 1676–1680 Vol.3, 2004.

[RO08] M. Raento and A. Oulasvirta. Designing for privacy and self-presentation in social awareness. *Personal Ubiquitous Comput.*, 12:527–542, October 2008.

[RPK+09] T. Raptis, C. Polychronopoulos, A. Kousaridas, P. Spapis, V. Gazis, N. Alonistioti, and I. Chochliouros. Technological enablers for self-manageable future internet elements. In *Future Computing, Service Computation, Cognitive, Adaptive, Content, Patterns, 2009. COMPUTATIONWORLD '09. Computation World:*, pages 499–504, 2009.

[RSO+04] L. B. Ruiz, I. G. Siqueira, L. B. e. Oliveira, H. C. Wong, J. M. S. Nogueira, and A. A. F. Loureiro. Fault management in event-driven wireless sensor networks. In *Proceedings of the 7th ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, MSWiM '04, pages 149–156, New York, NY, USA, 2004. ACM. Available at `http://doi.acm.org/10.1145/1023663.1023691`.

[Sai13] Sail Project. Scalable and adaptive internet solutions, 2010-2013. Available at `http://www.sail-project.eu/`.

[SAL06] J. Strassner, N. Agoulmine, and E. Lehtihet. Focale: A novel autonomic networking architecture. In *Latin American Autonomic Computing Symposium (LAACS)*, 2006.

[SBEKAP09] G. Sawma, R. Ben-El-Kezadri, I. Aib, and G. Pujolle. Autonomic management for capacity improvement in wireless networks. In *Proceedings of the 6th IEEE Conference on Consumer Communications and Networking Conference*, pages 683–688, Las Vegas, NV, USA, 2009. IEEE Press.

[SC05] H. Sabbineni and K. Chakrabarty. Location-aided flooding: an energy-efficient data dissemination protocol for wireless-sensor networks. *IEEE Trans. Comput.*, 54(1):36–46, Jan 2005.

[Sch96] J. Schoenwaelder. Using multicast-snmp to coordinate distributed management agents. *Systems Management Workshop, IEEE International*, 0:136, 1996.

[Sel11] Self-net Project. Self-management of cognitive future internet elements, 2011. Available at `https://ict-selfnet.eu/`.

[SFO+13] M. Simonin, E. Feller, A. Orgerie, Y. Jegou, and C. Morin. An autonomic and scalable management system for private clouds. In *Cluster, Cloud and Grid Computing (CCGrid), 2013 13th IEEE/ACM International Symposium on*, pages 198–199, 2013. 10.1109/CCGrid.2013.44.

[SGFT13] V. Sonoteas, L. Gkatzikis, P. Flegkas, and L. Tassiulas. Distributed cache management in information-centric networks. *Network and Service Management, IEEE Transactions on*, 10(3):286–299, 2013.

[SGR10] M. Subramanian, T. A. Gonsalves, and N. U. Rani. *Network Management: Principles and Practice*. Pearson Education India, 2010.

[SHP+10] M. Salmanian, J. Hu, L. Pan, P. Mason, and M. Li. Supporting periodic, strong re-authentication in manet scenarios. In *MILITARY COMMUNICATIONS CONFERENCE, 2010 - MILCOM 2010*, pages 19 –25, 31 2010-nov. 3 2010.

[SK08] S. Sohail and A. Khanum. Simplifying network management with fuzzy logic. In *Communications Workshops, 2008. ICC Workshops '08. IEEE International Conference on*, pages 195–201, 2008.

[SMLN+03] I. Stoica, R. Morris, D. Liben-Nowell, D. R. Karger, M. F. Kaashoek, F. Dabek, and H. Balakrishnan. Chord: a scalable peer-to-peer lookup protocol for internet applications. *IEEE/ACM Trans. Netw.*, 11(1):17–32, Feb. 2003.

[Sol11] SolarWind. Orion network performance monitor (npm), 2011. Available at `http://www.solarwinds.com/products/orion/`.

[SSC+13a] S. Sharma, D. Staessens, D. Colle, M. Pickavet, and P. Demeester. Automatic bootstrapping of openflow networks. In *Local Metropolitan Area Networks (LANMAN), 2013 19th IEEE Workshop on*, pages 1–6, 2013. 10.1109/LANMAN.2013.6528283.

[SSC+13b] S. Sharma, D. Staessens, D. Colle, M. Pickavet, and P. Demeester. A demonstration of automatic bootstrapping of resilient openflow networks. In *Integrated Network Management (IM 2013), 2013 IFIP/IEEE International Symposium on*, pages 1066–1067, 2013.

[ST05] M. Sbodio and W. Thronicke. Ontology-based context management components for service oriented architectures on wearable devices. In *Industrial Informatics, 2005. INDIN '05. 2005 3rd IEEE International Conference on*, pages 129–133, 2005.

[SVBM08] J. Singh, L. Vargas, J. Bacon, and K. Moody. Policy-based information sharing in publish/subscribe middleware. In *Policies for Distributed Systems and Networks, 2008. POLICY 2008. IEEE Workshop on*, pages 137–144, 2008.

[SWG+13] G. Sun, F. Wu, X. Gao, G. Chen, and W. Wang. Time-efficient protocols for neighbor discovery in wireless ad hoc networks. *Vehicular Technology, IEEE Transactions on*, 62(6):2780–2791, 2013. 10.1109/TVT.2013.2246204.

[Sys12] C. Systems. Cisco discovery protocol - cdp, Accessed in 2012. Available at `http://www.cisco.com/en/US/docs/ios/12_2/configfun/configuration/guide/fcf015.html#wp1004030`.

[SZN+07] S. Schuetz, K. Zimmermann, G. Nunzi, S. Schmid, and M. Brunner. Autonomic and decentralized management of wireless access networks. *Network and Service Management, IEEE Transactions on*, 4(2):96–106, 2007.

[TCS13] Y. Targali, V. Choyi, and Y. Shah. Seamless authentication across heterogeneous networks using generic bootstrapping systems. In *Wireless Communications and Mobile Computing Conference (IWCMC), 2013 9th International*, pages 137–142, 2013.

[TLG08] A. Tizghadam and A. Leon-Garcia. Aorta: Autonomic network control and management system. In *INFOCOM Workshops 2008, IEEE*, pages 1–4, 2008.

[Uni13] Univerself. Univerself project, 2010-2013. Available at `http://www.univerself-project.eu/about-univerself`.

[Vas06] S. Vasudevan. Self-organization in largescale wireless networks. Master's thesis, University of Massachusetts, Dept Computer Science, sep 2006. Available at `http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.115.4491&rep=rep1&type=pdf`.

[VCPP13] L. Valerio, M. Conti, E. Pagani, and A. Passarella. Autonomic cognitive-based data dissemination in opportunistic networks. In *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2013 IEEE 14th International Symposium and Workshops on a*, pages 1–9, 2013.

[VKT05] S. Vasudevan, J. Kurose, and D. Towsley. On neighbor discovery in wireless networks with directional antennas. In *Proc. IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies INFOCOM 2005*, volume 4, pages 2502–2512, 13–17 March 2005.

[W3C03] W3C. Extensible markup language (xml), http://www.w3.org/xml/, 2003.

[WBHW11] H. Wirtz, R. Backhaus, R. Hummen, and K. Wehrle. Establishing mobile adhoc networks in 802.11 infrastructure mode. In *Proceedings of the 6th ACM international workshop on Wireless network testbeds, experimental evaluation and characterization*, WiNTECH '11, pages 89–90, New York, NY, USA, 2011. ACM. Available at `http://doi.acm.org/10.1145/2030718.2030737`.

[Wea90] U. Warrier and et al. Common management information services and protocols. Technical report, RFC 1189, 1990.

[Wil08] L. Wilson. Distributed, heterogeneous resource management using artificial immune systems. In *Parallel and Distributed Processing, 2008. IPDPS 2008. IEEE International Symposium on*, pages 1–8, 2008.

[WLL10] J. Wang, Z. Li, and Y. Liu. Fractured voronoi segments: Topology discovery for wireless sensor networks. In *Mobile Adhoc and Sensor Systems (MASS), 2010 IEEE 7th International Conference on*, pages 137–145, 2010.

[WNZ+03] L. Wang, Z. Niu, Y. Zhu, H. Deng, and M. Yano. Integration of snr, load and time in handoff initiation for wireless lan. In *Personal, Indoor and Mobile Radio Communications, 2003. PIMRC 2003. 14th IEEE Proceedings on*, volume 3, pages 2032 – 2036 vol.3, sept. 2003.

[WS93a] C. Wang and M. Schwartz. Fault detection with multiple observers. *Networking, IEEE/ACM Transactions on*, 1(1):48–55, 1993.

[WS93b] C. Wang and M. Schwartz. Identification of faulty links in dynamic-routed networks. *Selected Areas in Communications, IEEE Journal on*, 11(9):1449–1460, 1993.

[XTL+10] F. Xu, C. Tan, Q. Li, G. Yan, and J. Wu. Designing a practical access point association protocol. In *INFOCOM, 2010 Proceedings IEEE*, pages 1 –9, march 2010.

[YB06] I. Yahia and E. Bertin. Towards autonomic management for next generation services. In *Networking and Services, 2006. ICNS '06. International conference on*, pages 38–38, 2006.

[YLGF08] C. Yu, A. Leon-Garcia, and I. Foster. Toward an autonomic service management framework: A holistic vision of soa, aon, and autonomic computing. *Communications Magazine, IEEE*, 46(5):138–146, 2008.

[YLH+11] L. Yuan, M. Liran, Y. Hongjun, C. Xiuzhen, C. Yong, A.-R. M. A., and A.-D. Abdullah. Load balancing access point association schemes for ieee 802.11 wireless networks. In *Proceedings of the 6th international conference on Wireless algorithms, systems, and applications*, WASA'11, pages 271–279, Berlin, Heidelberg, 2011. Springer-Verlag. Available at `http://dl.acm.org/citation.cfm?id=2033150.2033175`.

[YLWX10] D. Yan, J. Liu, B. Wang, and P. Xu. Topology control algorithm based on overlapping clustering. In *Networking, Sensing and Control (ICNSC), 2010 International Conference on*, pages 737–741, 2010.

[YWW99] J. Yang, P. Wang, and J. Wu. A scalable, web-based architecture for hierarchical network management. In *Global Telecommunications Conference, 1999. GLOBECOM '99*, volume 3, pages 1882–1888 vol.3, 1999.

[Zhi09] Z. Zhizhao. Distribution policy based information dissemination management system. In *Information Management, Innovation Management and Industrial Engineering, 2009 International Conference on*, volume 4, pages 325–328, 2009.

[ZHS+04] B. Y. Zhao, L. Huang, J. Stribling, S. C. Rhea, A. D. Joseph, and J. D. Kubiatowicz. Tapestry: a resilient global-scale overlay for service deployment. *IEEE J. Sel. Areas Commun.*, 22(1):41–53, Jan. 2004.

[ZLBY07] C. Zhongmin, J. Lincheng, X. Baowen, and W. Yeqing. The design and implementation of mobile agent-based network management system. In *Wireless Communications, Networking and Mobile Computing, 2007. WiCom 2007. International Conference on*, pages 3101–3105, 2007.

Annex A

Lightweight Discovery and Exchange of Network Information in Distributed Network Management

Lucas Guardalben, Tomé Gomes, Rui Sousa, Paulo Salvador and Susana Sargento

In IFIP/IEEE International Symposium on Integrated Network Management (IM), may 2013

The format has been revised.

Lightweight Discovery and Exchange of Network Information in Distributed Network Management

Lucas Guardalben, Tomé Gomes, Rui Sousa, Paulo Salvador and Susana Sargento

Abstract

Node cooperation is a key part of distributed management process, requiring periodic exchange of management information between the nodes. To increase scalability of the network information exchange, network nodes need to cooperate through an efficient set of rules, policies and criteria, to minimize the time and amount of control messages flowing in the network. We propose a solution for optimized decision of the network nodes to perform discovery and exchange of network information, through a Neighbors Eyesight Direction (NED) function and cooperation between the nodes. The proposed solution is embedded into a protocol, named HIde and SeeK Directional Decision (HISK2D), which is compared against current protocols for network discovery and awareness. We show, by means of experimentation that our HISK2D is more efficient than the base approaches analyzed, in terms of control overhead in a operator's link to discover the nodes, convergence time and exchange of network information according to the acquired knowledge of the network.

A.1 Introduction

Emerging networks, as envisioned in Next Generation Networks (NGNs), embrace multiple types of devices and communication technologies, e.g. Ethernet, Wi-Fi, 3G, 4G, etc. Moreover, management of large scale networks is an essential requirement, becoming a challenging situation from the operator's point of view. The operator needs to re-think the management techniques that will be used to respond quickly to this challenge with the purpose of increasing the quality of service for their customers.

This is only achievable through distributed management approaches which bring inherent support for self-management capabilities, enhancing the network with inbuilt intelligence. The current trend of distributed management requires the embedding of management functionalities inside the network nodes. However, this embedding requires network nodes to have sufficient information to take management decisions in an efficient and cooperative way.

Then, this information needs to be exchanged, which may increase the network overhead to unbearable limits. Therefore, new lightweight approaches need to be developed in order to organize the network nodes and determine the interactions between them, to enable optimized network information exchange. In order to deal with the large amount of information exchange, current distributed management solutions use traditional protocols for discovery, communication and cooperation that have difficulties to meet the requirements of low overhead and fast information exchange.

In response to this challenge, we propose the HIde and SeeK Directional Decision (HISK2D) protocol which comprises the several stages of communication, including the discovery of nodes and topology until the exchange of network information. This cooperation is provided by a Neighbors Eyesight Direction (NED) function, that enhances the discovery and exchange information beyond the limit of the directly connected nodes. The benefits provided by this directionality include choosing the best neighbor node to forward the discovery process and exchange of information according to the knowledge depth of the network (e.g. 1, 2 or 3 hops). The role of depth is to ensure the possibility to define the degree of knowledge of a node about its network neighborhood.

We compared the HISK2D protocol with three discovery solutions, namely: Cisco Discovery Protocol (CDP) [iCS12], Optimized Link State Routing (OLSR) [RFC03] and the discovery approach of Open Shortest Path First (OSPF) [Moy98] over an experimentation scenario varying from 4x4 until 9x9 nodes on a grid topology. The results show that the HISK2D is more efficient in terms in terms of control overhead in a operator's link to discover the nodes, convergence time and exchange of network information according to the acquired knowledge of the network (e.g. directed connect nodes or depth larger than 1 hop).

The remainder of the paper is organized as follows. The Section A.2 describes the HISK2D protocol and the implemented modules and their interactions. Section A.3 presents the experimental results. Finally, Section A.4 concludes the paper.

A.2 HIde and SeeK Directional Discovery: HISK2D Protocol

The HISK2D protocol was proposed to optimize the discovery process and the amount of information exchanged between the nodes, enabling nodes cooperation in different points of the network. The discovery and exchange of network information in HISK2D are complementary processes: initially, the discovery aims to find the nodes according to the requested knowledge of the network in number of hops (e.g 1, 2, 3...); then, the information gathered through the discovery process is exchanged between the nodes at 1-hop distance, or also through the next hop relay nodes. As part of the nodes discovery mechanism of HISK2D, it was proposed the Hide and Seek concept [GGP+11].

The interactions in the Hide and Seek discovery (see Figure A.1) are made at 1 hop distance in this initial stage, and they also exchange information about each other, such as identifiers (MAC and Internal control identifier), network status, local resources available (% of free CPU and Memory), source and destination IP addresses. All gathered information is recorded in a local repository of each *Seeker* node. A node in *Hider* mode (e.g N13) waits for a *Seeker* contact message (or it waits a random time in case it does not receive any message), and then becomes a new *Seeker* cooperating to discover and exchange information with other *Seeker* or *Hider* nodes. This process is repeated until all nodes have been contacted.

However, the information at 1 hop distance may not be sufficient for an efficient distributed management process, since it may benefit from the knowledge of a larger neighborhood. To optimize the amount of information exchanged with low overhead and time at several hops distance, HISK2D uses a Neighbors Eyesight Direction (NED) function, which aims to narrow the directions through which direct nodes or next hop relay nodes will be chosen to continue the discovery and exchange of information.

The NED also brings benefits to distributed management process due the possibility to control the amount of information to be discovered and exchanged in the network, by enabling the information cooperation between nodes. Therefore, this process depends on relay nodes to forward messages until the depth limit is reached. The role of depth (in number of hops) is to ensure the possibility to define how much a node wants to know about the network, and to decide who is the best neighbor to become the next hop relay node in this depth knowledge. Regarding the difference of the proposed solution with the previous works, this is the one that integrates the NED decision function and the depth, which will provide the partial view (see Equation A.2) with different depths.

The criteria used to choose the next hop neighbor is based on the ranking of the gathered information from the nodes at 1 hop distance. The nodes are also able to maintain the information larger than 1 hop in the repository. The NED decision is based on the link capacity, local node resources available (% of free CPU and Memory) and also nodes density as input parameters, aiming to choose only the 1 hop candidate node that will continue the discovery process, keeping the information gathered between the relay nodes chosen as well. In order to show the NED decision criteria, we define $\mathscr{A}$ as the complete set of nodes in the scenario:

$$\mathscr{A} = \{i : i = 1, \dots, N\} \tag{A.1}$$

In any given time instant, all nodes have a local repository named partial view of the network, i.e., the set of nodes discovered from a defined hop distance. The partial view of node i can be defined by:

$$\mathscr{P}_{i,d} = \{j : j \in \mathscr{A} \wedge j \neq i \wedge v(i,j) = d\} \tag{A.2}$$

where $v(i,j)$ is a function that returns the hop distance (d) between node i and j if there is a path between them, or 0 otherwise.

The link capacity is given by:

$$\mathcal{LC}_i = \sum_{j \in \mathscr{P}_{i,1}} \mathcal{L}_{i,j} \tag{A.3}$$

where $\mathcal{L}_{i,j}$ represents the reference value of the link capacity $(\mathcal{L})$ from node i to j. This is determined by the interface type and/or software configuration.

Local and network resources are given by:

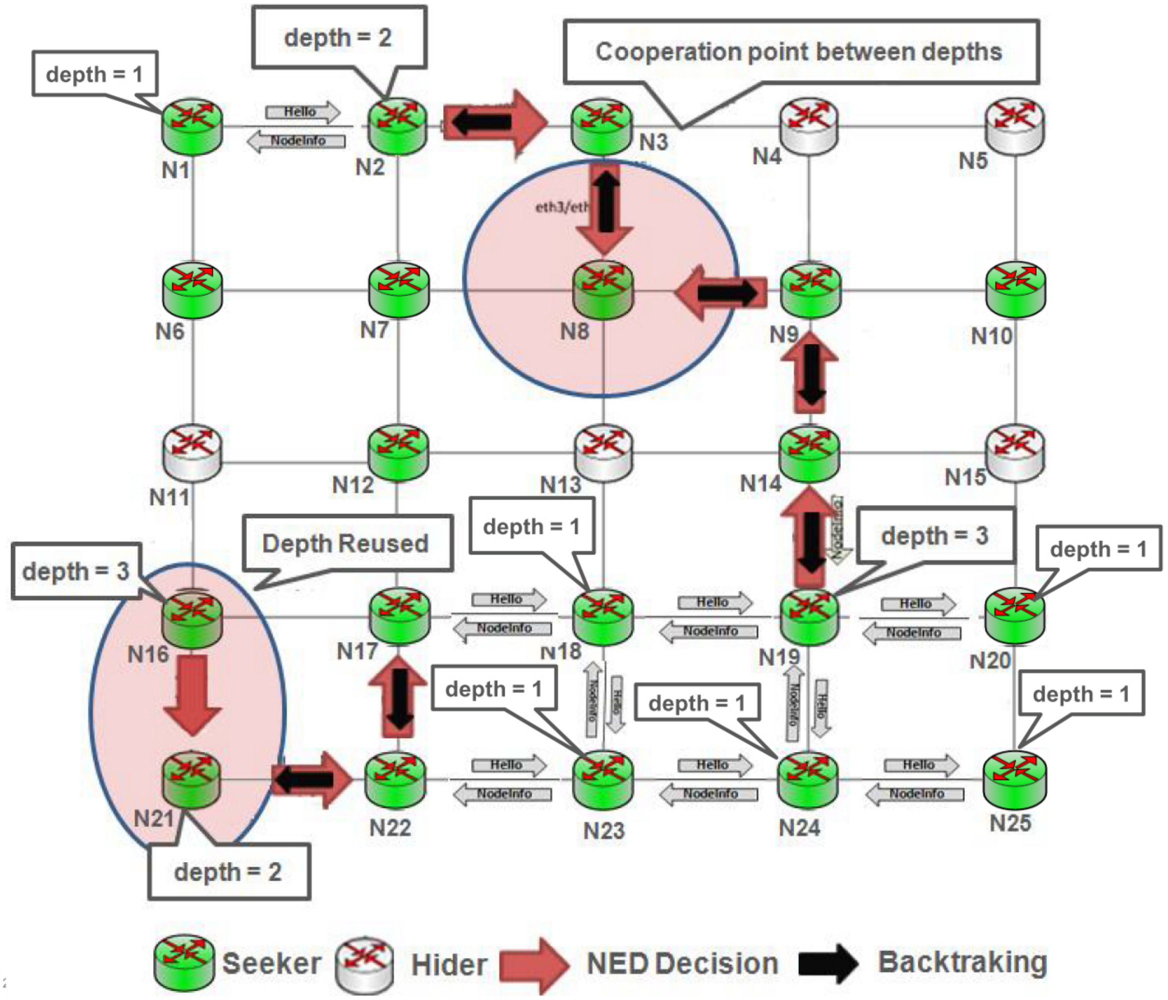

Figure A.1: Example of Hide and Seek and Directional Discovery process using different depths through the Neighbors Eyesight Decision(NED)

$$\mathcal{R}_i = \sum_{j \in \mathscr{P}_{i,1}} (\mathcal{C}_j + \mathcal{M}_j) * \mathcal{T}_{i,j} \tag{A.4}$$

where $\mathcal{C}_i$ represents the available CPU, $\mathcal{M}_i$ the available RAM Memory, and $\mathcal{T}_i$ represents the lowest RTT (Round-Trip-Time) measured by node i communicating to node j.

Network nodes density is given by:

$$\mathcal{N}_i = max|\mathscr{P}_{j,1}|, \forall j \in \mathscr{P}_{i,1} \wedge j \neq i \tag{A.5}$$

where $max|\mathscr{P}_{j,1}|$ corresponds to the largest Partial View of all connected neighbors, and $|.|$ represents the cardinality of a set.

The global value of the NED is obtained by performing a weighted sum, given by:

$$\mathcal{NED}_i = w_L \mathcal{LC}_i + w_R \mathcal{R}_i + w_N \mathcal{N}_i \tag{A.6}$$

where, w_L, w_R and w_N represent the weights of (A.3), (A.4) and (A.5). Moreover, the maximum value returned will determine the best neighbor chosen by the node i. Note that each node $\boldsymbol{i}$ will only consider the neighbors $\boldsymbol{j} \in \boldsymbol{P_{i,1}}$ to determine the NED global value (A.6). Focusing on the factors, was considered the absolute values for link capacity, RTT and the maximum number of links and interfaces. Additionally, the factors respecting to local node resources, like CPU and RAM memory were normalized, e.g. percentage of CPU and RAM available.

In Figure A.1, by default *Seeker* nodes have knowledge at depth 1 hop, being also possible to configure a node with depth larger than 1 hop (e.g N2, N16 and N19). As an example, the initial depth of N19 is 3, and all nodes directly connected to N19 will send their depth and knowledge (partial view) to N19. Then, the NED function will choose a 1-hop neighbor according to the criteria presented in the equation (A.6) to continue the discovery and exchange of the already obtained information to this chosen node. Then, the best node to continue the discovery is chosen to be N14, and information about N18, N20 and N24 will be exchanged to N14. Node N14 will repeat the 1-hop discovery and will choose a new direction (e.g N9). At this point, node N14 will have information about discovered nodes N9, N18, N20, N24, N19. Then, the next hop chosen is N9, and information about all nodes discovered at this point, plus the ones discovered from N9 (e.g N10, N14) will be sent to the chosen node N8, which reached the depth limit.

All information collected from N19 to N8 will back track to all nodes that belong to the NED paths (e.g N9, N14) up to the node that originated the request (e.g N19). The NED function process ends when: (1) the depth limit defined previously is achieved; or (2) all possible paths between the nodes are explored by the NED function. Note that it is also possible to configure different depth requests in the scenario. For example, consider N2 with depth equal to 2. Then, N2 will send all the gathered information from the nodes N1, N7 to the chosen node N3, and then, node N3 will repeat the process to the next chosen node N8, which reached the depth limit. Node N8 will back track the information gathered up to the node that originated the depth request (e.g N2).

In the case that different NED paths converge to the same node (e.g. N8), this node is considered to be a cooperation point and will back track all information gathered, e.g. from the paths with depths at 2 and 3, to the nodes that originated the requests (N2 and N19). After this process, both nodes (N2 and N19) will synchronize and update the partial views with their directed connected nodes, containing all nodes and its gathered neighborhood hop-by-hop from the NED path direction (e.g N3, N8, N9, N14). After that, all directed nodes already synchronized are able to exchange and cooperate according to a new requested depth.

In addition, in any given time, there may be multiple cooperation points in the network, increasing the efficiency of information cooperation process. Moreover, the NED function can also reuse the knowledge from a node (e.g. N21), without making any further demands. For instance, N16 requests the knowledge with depth equal to 3, and N21 already has this information; then, N21 can reuse the information previously gathered in order to reply to N16. In order to maintain the information always updated, the partial views of each node consider the most recent information exchanged as well.

A.3 Experimentation Results

A.3.1 Virtual Grid Scenario Set-up

The scenario was initially set up as virtual grid testbed containing OpenWrt Bleeding Edge r28129 (Guest Xen paravirtualized) as virtual machines on a HP Proliant server (CentOS-5 kernel Xen). The testbed is created through a Python script which automatically generates the virtual bridges and link connections between the virtual machines. Experiments are carried out for networks in grid topology with 4x4 (16 machines), 6x6 (36 machines), 8x8 (64 machines) and also 9x9 (81 machines). The results are obtained in two different scenarios: in the first one, the objective is to discover information only from the directly connected nodes; in the second, it is considered the information from neighbors at different depths: 1, 2 or 3 hops. In both scenarios, the interval between periodic packets (e.g. Hello) was configured to be 10 seconds.

In HISK2D, this interval is dynamically adapted with a mean around 10s. The values presented in the graphs below are an average of 5 repetitions and 95% of confidence interval. To perform the packets capture, TCPdump is the software used in each node interface, with an observation time of 10 minutes. The links capacity is considered to be 1 Mb/s. We compare our approach with other well known discovery baselines, using open-source versions of OSPF version 2, CDP version 2 and OLSR version 4. The use of OLSR as one of the baseline protocols is due to the fact that its discovery concept considers complete information of the network (it can also be used in wired networks). This allows to compare the advantages of having the partial or complete information. Moreover, all solutions evaluated share the same events and conditions regarding the programming techniques.

A.3.2 Results

The impact of the overhead in a typical operator link is measured and it is shown in Figure A.2. Assuming a link capacity of 1 Gbit/s, the percentage of overhead is very low for all protocols. However, the values of the OLSR overhead are very high compared with the other protocols. Once more, Figure A.2 is presented with a zoom to the other three protocols. As it can be noticed, the impact of the CDP protocol grows proportionally with the network size, and OSPF protocol has a behavior similar to OLSR but with smaller impact.

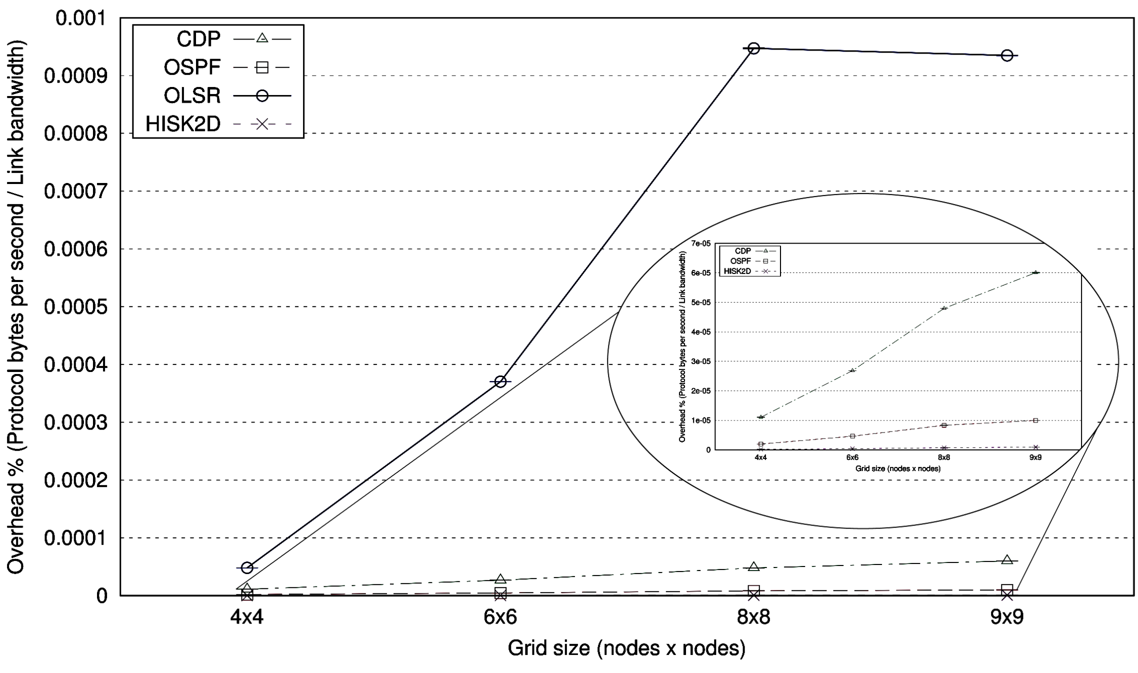

Figure A.2: Overhead using 1Gbit/s link as reference

Regarding the convergence time, OSPF protocol tends to require more time to discover its neighbors, as shown in Figure A.3. This was expected, since for large networks, the calculation of Dijkstra's algorithm is complex and consumes significant time. CDP and OLSR protocols have convergence times relatively similar, but comparatively to HISK2D, they all require more time to achieve the same result, i.e., using HISK2D each node discovers more quickly their directly connected neighbors. The chosen signaling mechanism, based on request/response, internal triggers and events, is responsible for this quick performance.

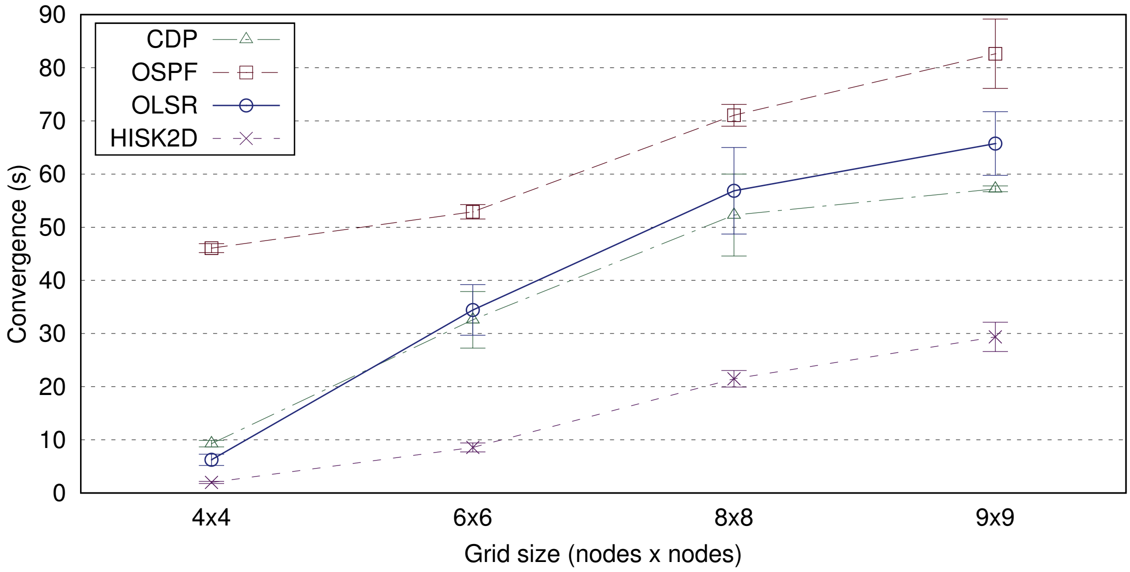

Figure A.3: Discovery convergence time

Figures A.4 to A.5 analyze the amount of information exchanged, the number of nodes discovered and the average convergence time for the second scenario, according to the configured depth. The experiments are carried out in a network with 81 virtual nodes in grid topology. The NED weights are considered to be 33% for each one: link capacity (w1), local node and network resources (w2) and number of interfaces (w3).

This analysis is performed on each machine varying the depth of the discovery process from 1 to 3 hops. In HISK2D, this is directly configured in NED function, which will determine the best interface through which the discovery should proceed. With OSPF, the depth is set using network areas: each node will have its interfaces configured to belong to a certain area depending on the desired depth, i.e., higher values of depth imply larger OSPF network areas.

Figure A.4 shows that HISK2D requires less packets than OSPF, regardless of the chosen depth. This is mainly justified by the different roles in HISK2D: when the node is *Hider*, it waits to be contacted or starts itself the discovery process after a certain period of time. This is also reflected in the lower amount of data exchanged, although not shown here due to limitations.

This does not necessarily mean that HISK2D has less information than OSPF, but rather that no redundant data is used, i.e., only the strictly required one is forwarded between nodes. As an example, periodic updates only include the parameters that are changed in the meantime.

To prove that the previously described behavior does not affect the quickness of the HISK2D discovery mechanism, the convergence time is presented in Figure A.5. Setting any of the presented depth values, HISK2D is, at least, 60% faster than OSPF.

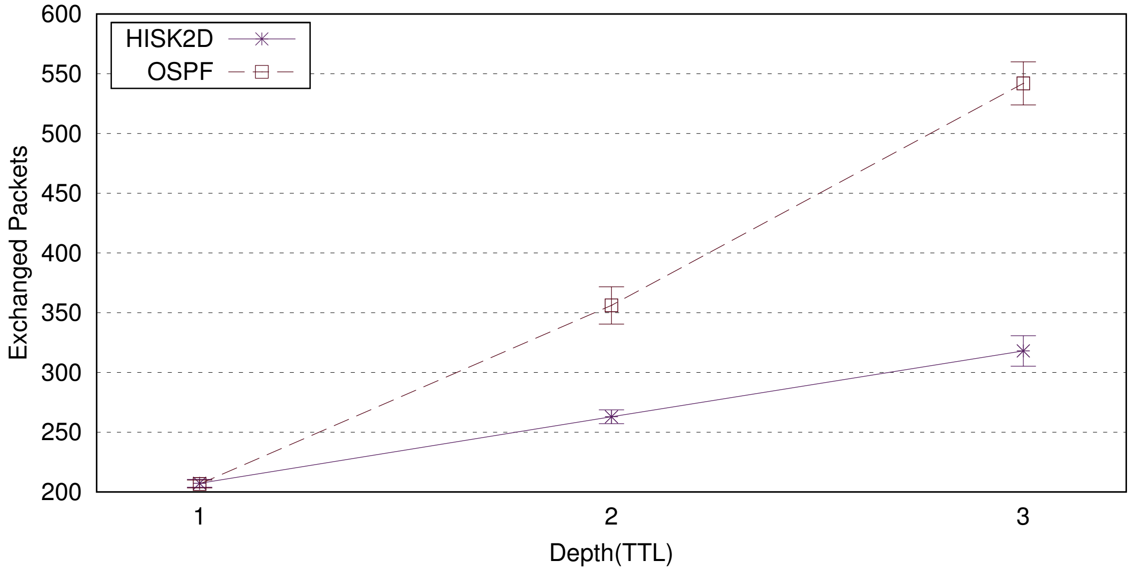

Figure A.4: Grid 9x9: Number of Exchanged Packets

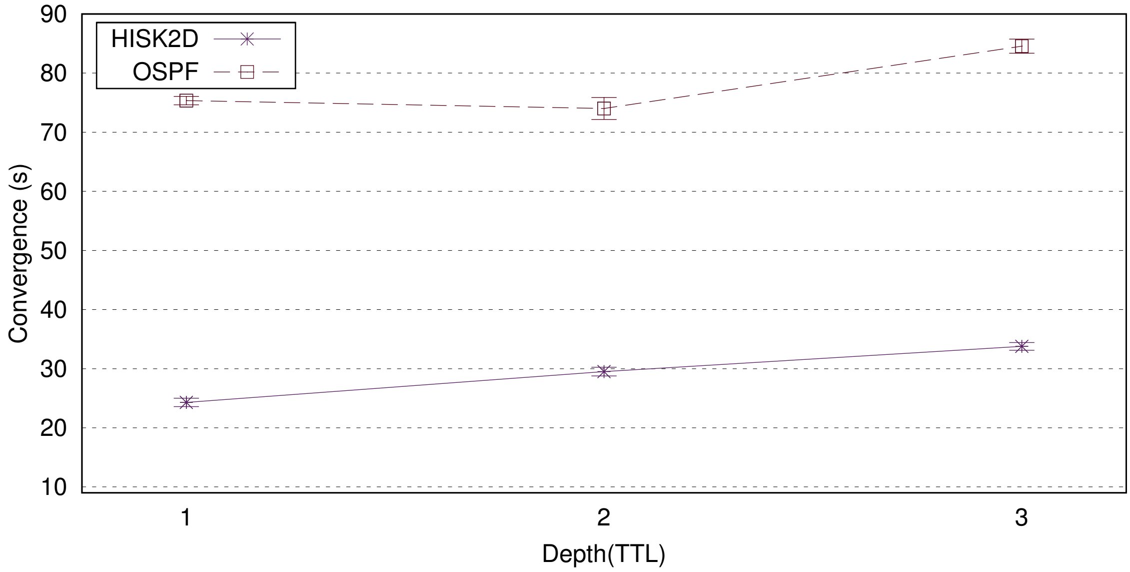

Figure A.5: Grid 9x9: Convergence Time in Seconds to Discovery the Nodes

A.4 Conclusions and Future Work

This paper presented HISK2D, a novel protocol for discovery and exchange of network information in large-scale distributed network management. The results present the benefits provided by the HISK2D in terms of control overhead in a operator's link to discover the nodes, convergence time and exchange of network information according to the acquired knowledge of the network. Moreover, the HISK2D proves to be scalable for large-scale scenarios, and also enforces the nodes to search the best neighbor direction through the NED criteria, contributing to improve resource aggregations and reducing the required signaling messages as well. This contributes to a high efficiency in distributed network management. As future work, we plan to adjust the NED weights in real time to optimize the number of discovered nodes or other relevant criteria, according to the network context. We also plan to evolve HISK2D to wireless scenarios.

A.5 Acknowledgments

This work was supported by the Fundação para Ciência e Tecnologia - FCT, through the grant SFRH/BD/62511/2009. The authors also thank the support provided by UBIQUIMESH project PTDC/EEA-TEL/105472/2008.

References

[GGP+11] L. Guardalben, T. Gomes, A. Pinho, P. Salvador, and S. Sargento. Nodes discovery in the in-network management communication framework. In *MONAMI*, pages 145–157, 2011.

[iCS12] inc Cisco Systems. Cisco discovery protocol - cdp, 2012. http://www.cisco.com/en/US/tech/tk648/tk362/tk100/tsd_technology_support_sub-protocol_home.html.

[Moy98] J. Moy. *OSPF Version 2*. IETF RFC 2328, April 1998.

[RFC03] RFC 3626, United States. *Optimized Link State Routing Protocol (OLSR)*, 2003.

Annex B

On the Analysis of Dissemination Management Information through an Eyesight Perspective

Lucas Guardalben, Tiago Condeixa, Tomé Gomes, Paulo Salvador and Susana Sargento

In Globecom 2013 Workshop - Management of Emerging Networks and Services (GC13 WS - MENS)

The format has been revised.

On the Analysis of Dissemination Management Information through an Eyesight Perspective

Lucas Guardalben, Tiago Condeixa, Tomé Gomes, Paulo Salvador and Susana Sargento

Abstract

The dissemination of information in wireless networks is a fundamental aspect in several network processes, such as routing, monitoring and management. Nowadays, most of the techniques for dissemination of management information adopt controlled flooding approaches, which usually impairs the network performance. In this sense, the optimization of the dissemination of management information for dense wireless scenarios remains a relevant research challenge. In this paper, it is presented and analyzed the performance of the Neighbors Eyesight Direction (NED) dissemination against to OSPF, uncontrolled flooding and Gossip approaches in wired and large network scenarios. NED is able to work on wired and dense wireless scenarios, showing to be more efficient than the baseline approaches. With lower dissemination cost and lesser recursive cycles, NED proved to be scalable to disseminate management information for scenarios with a large number of devices.

B.1 Introduction

In wireless networking environments, as envisioned in future networks, are showing an exponential growth in the number of network devices, as well as demanding mobile applications and consumers of data information. The large amount of information to be disseminated is an important challenge from the management point of view. In fact, the management information needs to be disseminated to a larger set of devices without compromising the network performance. The current approaches to disseminate management information suffer from common problems, such as: uncontrolled flooding, high level of redundancy messages, energy and device's resources consumption. To ensure that the information is able to be disseminated in large areas, the devices need to be aware of the surrounding devices context, in order to select the best node to relay the dissemination process/information. With this respect, new mechanisms and techniques shall efficiently provide dissemination of information, taking into account the necessity of collaboration and cooperation among devices.

This paper provides a comparative analysis of the Neighbors Eyesight Direction (NED) dissemination, against three solutions already extensively studied in the literature [Tan02], uncontrolled flooding and Gossip and OSPF [Moy98]. This analysis is performed over wired, random and realistic dense wireless scenarios. The obtained results show that the NED dissemination is able to work on wired and dense wireless scenarios, showing to be more efficient than the baseline approaches, both in terms of dissemination cost and recursive cycles. Additionally, NED also brings benefits to distributed management process, due to the possibility to control the amount of information to be disseminated in the network, by enabling the cooperation between devices.

The remainder of the paper is organized as follows. The related work is described in the Section B.2. Subsequently, Section B.3 briefly describes the Neighbors Eyesight Direction dissemination. Section B.4 performs the analysis and comparison of the results. Finally, Section B.5 concludes the paper.

B.2 Related Work

The propagation of information is fundamental to most of the network processes, such as routing, monitoring and management. There is a wide set of approaches in the literature for the exchange and dissemination of information in the network, most of them based on variants of the following approaches: flooding-based [CHLS07]; probability-based [DFKS07] to reduce the amount of dissemination messages; MCDS-Based [Meg06] to define a sequence of stable connected dominating sets; location-based [SCM05] to disseminate only to specific locations; epidemic-based [Gen07] to opportunistically disseminate through the overall network; and cluster-based [CMS08] to limit the dissemination inside network clusters. Furthermore, in most of these cases, the approaches use central managers to perform the exchange of information. The authors in [DDL+02] implemented a policy management dissemination in a hierarchical domain. Each policy is a java object working in a hierarchical domain, in order to maintain the policy attributes, while the policy control object is created to coordinate run-time access that disseminates each policy in the network. This approach uses a central coordinator, which may not be compliant with scalability requirements. For policy based information dissemination, [Zhi09] proposes an Information Dissemination Management, which controls all steps of the exchanging of information. Therefore, management of policies in dense systems is extremely complex, because of the potentially large number of policies that need to be disseminated. Some management dissemination techniques were developed integrating the use of ontologies [Zhi09, ST05, DBS+07] to organize the information and to disseminate it between the devices, becoming useful for large information content. However, the ontology-based approaches need a well-defined semantic classification to perform the correct dissemination of the information, which is difficult to achieve in dense scenarios.

Consequently, the presented techniques are not efficient to perform dissemination for distributed wireless network management, motivating this research direction proposed and evaluated in this paper.

B.3 An Eyesight Perspective

Neighbors Eyesight Direction (NED), firstly presented in [Gea], is a dissemination function which aims to narrow the directions through which neighbor devices (acting as end-points or relays) are chosen to continue the dissemination of management information. This process depends on relay devices to forward dissemination messages until the depth limit is reached. The depth is the number of hops to disseminate the information, which ensures the possibility to define how long a device wants to disseminate the management information in the network. For discovery process, the Hide and Seek (H&S) concept was proposed and integrated to work with the NED function [Gea, GGP+11], in order to create and gather information of surrounding neighbors. The work previously developed in [Gea] considers the NED applicability only in static wired and small scenarios. Therefore, this paper performs a deep and throurough evaluation of the proposed approach, against baseline approaches, in dense and random wireless scenarios.

The criteria used to choose the next hop neighbor to continue the dissemination process is based on the calculated ranking from the gathered information from neighbors devices [Gea]. For wireless scenarios, the NED decision is based on the parameters, such as link bandwidth available, network and device's resources available (RTT, CPU and Memory) and by the neighbors devices density. The link bandwidth available parameter, induces the NED to find better link conditions to disseminate the information, e.g., congestinoned links or lower link available bandwith are not suitable to flow the dissemination process.

In case of local and network resources, the NED induces the choice of neighboring devices that have good resources and network conditions to relay the management information, e.g, if a device has high RTT, no free CPU or memory, it is bounded to be a good relay information device. At last, the network devices density parameter induces the NED to choose neighbors that have more density of nodes, e.g., if a neighbor has fewer nodes in its surrounding area, thus, it is not probably a good choice to disseminate the management information to a large number of nodes.

Thus, the NED chooses only the best candidates device through eyesight direction, which is given by the set of the following equations.

Link bandwidth available:

$$\mathcal{BA}_i = \sum_{j \in \mathscr{P}_{i,1}} \mathcal{B}_{i,j} \tag{B.1}$$

where $\mathcal{B}_{i,j}$ represents the reference value of the bandwidth available ($\mathcal{B}$) from device i to j. This can be determined by packet pairs/train techniques as suggested by [ML08]. $\mathscr{P}_{i,1}$, is the local information about the neighbors named partial view, i.e., the set of neighbor devices discovered by the device i.

The partial view of device i can be defined by: $\mathscr{P}_{i,d} = \{j : j \in \mathscr{N} \wedge j \neq i \wedge v(i,j) = d\}$, where $v(i,j)$ is a function that returns the hop distance (d) between device i and j if there is a path between them, or 0 otherwise. $\mathscr{N}$, is the complete set of devices in the scenario, where, $\mathscr{N} = \{i : i = 1, \ldots, n\}$.

Local and network resources:

$$\mathcal{R}_i = \sum_{j \in \mathscr{P}_{i,1}} (\mathcal{C}_j + \mathcal{M}_j) * \mathcal{T}_{i,j} \tag{B.2}$$

where $\mathcal{C}_i$ represents the available CPU, $\mathcal{M}_i$ the available RAM Memory, and $\mathcal{T}_i$ represents the lowest RTT (Round-Trip-Time) measured by device i communicating to device j.

Network devices density:

$$\mathcal{N}_i = max|\mathscr{P}_{j,1}|, \forall j \in \mathscr{P}_{i,1} \wedge j \neq i \tag{B.3}$$

where $max|\mathscr{P}_{j,1}|$ corresponds to the largest Partial View of all connected neighbors, and $|.|$ represents the cardinality of a set.

The global value of the NED is obtained by performing a weighted sum, given by:

$$\mathcal{NED}_i = w_1 \mathcal{BA}_i + w_2 \mathcal{R}_i + w_3 \mathcal{N}_i \tag{B.4}$$

where, w_1, w_2 and w_3 represent the weights of (B.1), (B.2) and (B.3). The maximum value returned will determine the best eyesight direction neighbor chosen by the device i. Note that each device i will only consider the neighbors $j \in \mathscr{P}_{i,1}$ to determine the NED global value (B.4).

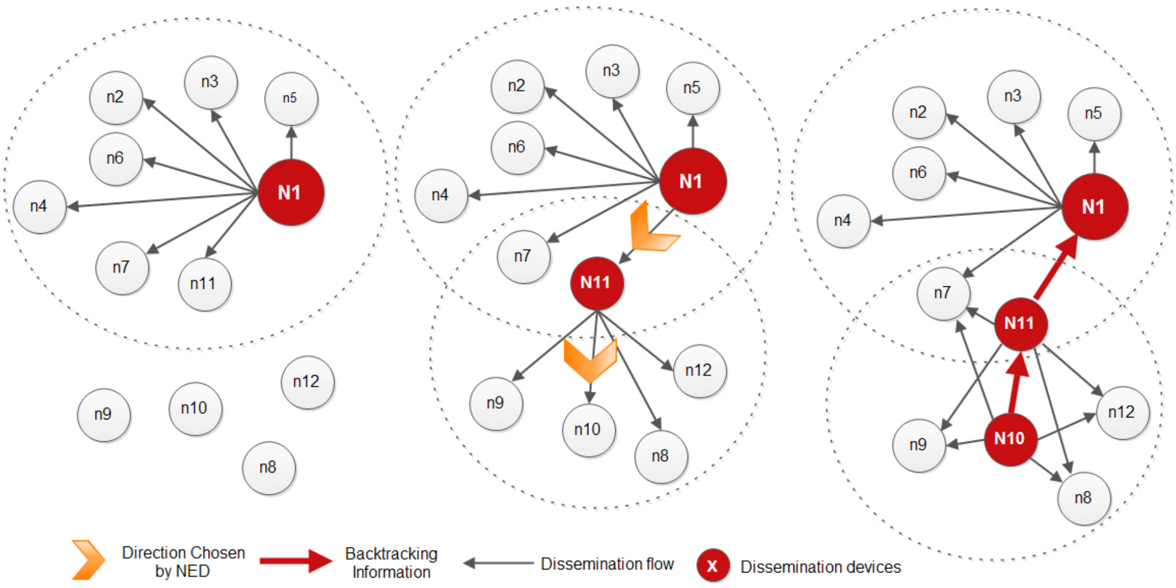

Figure B.1: Basic operating principles of NED

In order to demostrate the basic operating principles of the NED, in Figure B.1, $N1$ is considered a dissemination device, where it is started the dissemination process to the neighbor devices $n2, n3 \ldots n11$. All devices directly connected to $N1$ send and receive the management flow informations. We consider management information as all type of data gathered from devices that compose the network, such as: identifiers, network topology, devices' resources status, bandwidth available, delays, network density, network addresses and domains. Then, the direction chosen by the NED function is based on the maximum rank of the Equation

B.4 to continue the dissemination process. So, the best direction chosen to continue the dissemination is the device $\boldsymbol{N11}$, and information about $\boldsymbol{N1}$ and its neighbors is disseminated to $\boldsymbol{N11}$. Note that, the density of the $\boldsymbol{N11}$ and $\boldsymbol{N7}$ are the same, but $\boldsymbol{N11}$ probably has better bandwidth available and local/network resources than $\boldsymbol{N7}$, thus, the NED opted to choose $\boldsymbol{N11}$ as the relay node.

The device $\boldsymbol{N11}$ repeats the process and chooses a new direction (e.g $\boldsymbol{N10}$). After this process, the device $\boldsymbol{N11}$ also disseminates the management information to the next hop chosen, and finally all the information from $\boldsymbol{N1}$ is disseminated through the neighbor devices. All information collected from $\boldsymbol{N10}$ to $\boldsymbol{N11}$ is back tracked to all devices that belong to the NED paths (e.g. $\boldsymbol{N11}$) up to the device that originated the request (e.g. $\boldsymbol{N1}$).

The NED function process ends when: (1) the depth limit previously defined is achieved; or (2) all possible paths between the devices are explored by the NED. Note that it is possible to configure different depth requests for each dissemination device. The NED can also reuse the depth from a device, without making any further demands.

B.4 On the Analysis and Results

Dissemination of management information is one of the major challenges in dense wireless scenarios. The distributed management approaches are characterized by completely eliminating the centralized coordination device, being crucial to assess how information is disseminated on a network built with devices, acting as both relays and end-points. We will present both real experiments and simulation results to assess the performance of the NED dissemination approach.

B.4.1 Experimentation Results

The scenario was initially set up as a virtual testbed containing virtual machines (Guest Xen paravirtualized) on a HP Proliant server (CentOS-5 kernel Xen). The testbed is created through a Python script which automatically generates the virtual bridges and link connections between the virtual machines.

The experiments are carried out in a network with 81 virtual nodes in static grid topology. The NED weights are considered to be 33% for each one: link capacity (w1), local node and network resources (w2) and number of interfaces (w3). This analysis is performed on each machine varying the depth of the NED dissemination from 1 to 3 hops, and considering all nodes as dissemination points. We compare the NED dissemination with the dissemination mechanism of OSPF version 2. We consider as management information: number of nodes, ID/MAC, IP, RTT, Number of interfaces, CPU and memory available, bandwidth and number of hops. With OSPF, the depth is set using network areas: each node will have its interfaces configured to belong to a certain area depending on the desired depth, i.e., higher values of depth imply larger OSPF network areas.

The results are obtained considering the applicability of the NED function to disseminate information at different depths: 1, 2 or 3 hops. The values presented in the graphs below are an average of 5 repetitions and 95% of confidence interval. To perform the packets capture, TCPdump is the software used in each node interface, with an observation time of 10 minutes. The links capacity is considered to be 1 Mb/s.

Figures B.2 and B.3 analyze the cost to discovery/disseminating management information and the average number of disseminated nodes, according to the configured depth.

Figure B.2 shows that NED requires less bytes than OSPF, regardless of the chosen depth. This is also reflected in the lower amount of data exchanged to disseminate the management

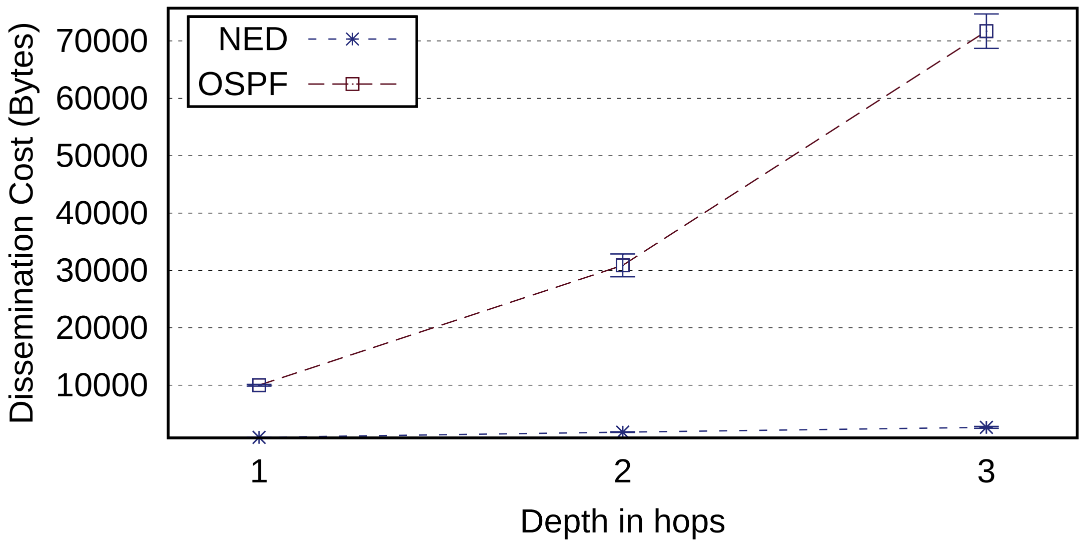

Figure B.2: Cost in bytes to disseminate management information

information. This does not necessarily mean that NED has less information than OSPF, but rather that no redundant data is used, i.e., only the strictly required one is forwarded between nodes. As an example, periodic updates only include the parameters that are changed in the meantime.

Figure B.3 shows that the average number of disseminated nodes by each node is similar using both protocols. Therefore, NED can achieve the same results of OSPF using less network resources regardless of the desired depth. This is due to the fact that NED function is enabled when the depth is larger than 1 hop, causing the dissemination process to follow the chosen paths. When the flow reaches the predefined depth, nodes crossed by the flow are unable to send update messages until the backtracking process has been finished. This behavior brings a lower overhead to the network communication.

B.4.2 Simulation Results

The behavior of the NED dissemination is also analyzed with MATLAB [MAT11] simulations, in dense static wireless networks, from 100 up to 500 devices, placed with random and realistic models. The wireless transmission range of the devices was defined with 100m. The random model places the devices from the uniformly distributed MATLAB function, in the simulation area of 500m x 500m. The realistic model adopted for devices placement is based on the topology type NPART/Berlin (from NPART [BM09]), which simulates the placement of the wireless devices in the city of Berlin, adjusting the simulation area (500m x 500m).

The uncontrolled flooding and Gossip [Mat05] approaches are used as the baselines, since they are the most adopted approaches for comparison purposes in the literature. The uncontrolled flooding implementation is based on [Mat05], but without flooding control. The evaluation is performed in a area of 500m x 500m, changing the number of devices and the number of dissemination devices, where it is ensured that none device in the network is isolated.

The number of dissemination devices, which is the number of devices that would start the dissemination process, is defined as 10% of the total number of devices in the scenario (e.g.

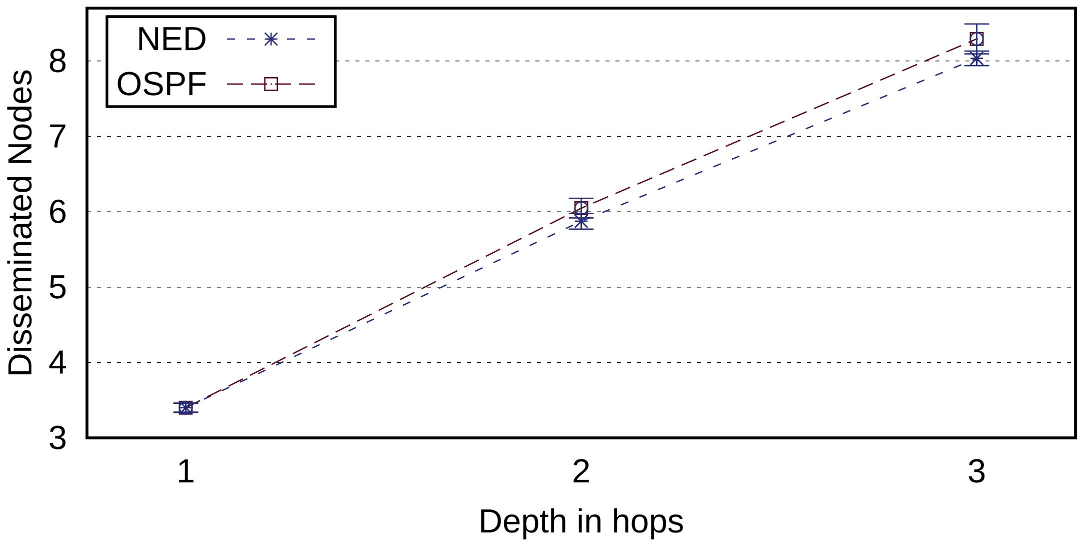

Figure B.3: Disseminated Nodes

300 devices has 30 dissemination points), uniformly distributed through these devices. Notice that the number of dissemination devices is performed only for NED and Gossip, since in the uncontrolled flooding all the devices act as dissemination devices.

In the Gossip, the maximum number of neighbors to performed the flooding dissemination is 25% of the dissemination devices, in order to avoid unnecessary and uncontrolled flooding (e.g. for 20 dissemination devices the Gossip probabilistically chooses only 5 neighbors to flood the management information).

The NED results are obtained from different experimental-driven combinations (111 | 100 | 010 | 001) of the weights $\boldsymbol{w1}$, $\boldsymbol{w2}$ and $\boldsymbol{w3}$ (see section B.3), and for a dissemination's depth much higher than the number of wireless devices. The results are obtained from 10 repetitions with different seeds and a confidence interval of 95%.

For the evaluation of the three analyzed approaches, three metrics are proposed, dissemination cost, recursive cycles, and disseminated devices:

- **Dissemination cost** is defined as $\boldsymbol{Dc = 1 + \sum nsize}$, where 1 is the broadcast discovery messages and $\boldsymbol{nsize}$ is the number of neighbor devices in which the management information is disseminated. The redundancy messages are also considered, but it depends on each algorithm (e.g. uncontrolled flooding has a higher probability to have redundancy messages, since all devices perform flooding). In terms of the analysis, lower dissemination cost is better.

- **Recursive cycles** measure the impact in terms of local device resources consumption to execute the algorithm. When the algorithm recursively sends broadcast messages to the neighbor devices, it is counted as 1 cycle. The lower number of recursive cycles means less device resources consumption (e.g. CPU and Memory).

- **Disseminated devices** measures the number of devices that receive the dissemination messages. The higher number of disseminated devices is better.

B.4.3 Dissemination Cost

NED reduces the dissemination cost when compared with Gossip and uncontrolled flooding, for both random (Figure B.4) and realistic (Figure B.5) scenarios, especially with the growth of the number of devices in the network (e.g. up to 200). The uncontrolled flooding (black line) and Gossip (green line) have a higher cost in both scenarios to disseminate the same management information in the network.

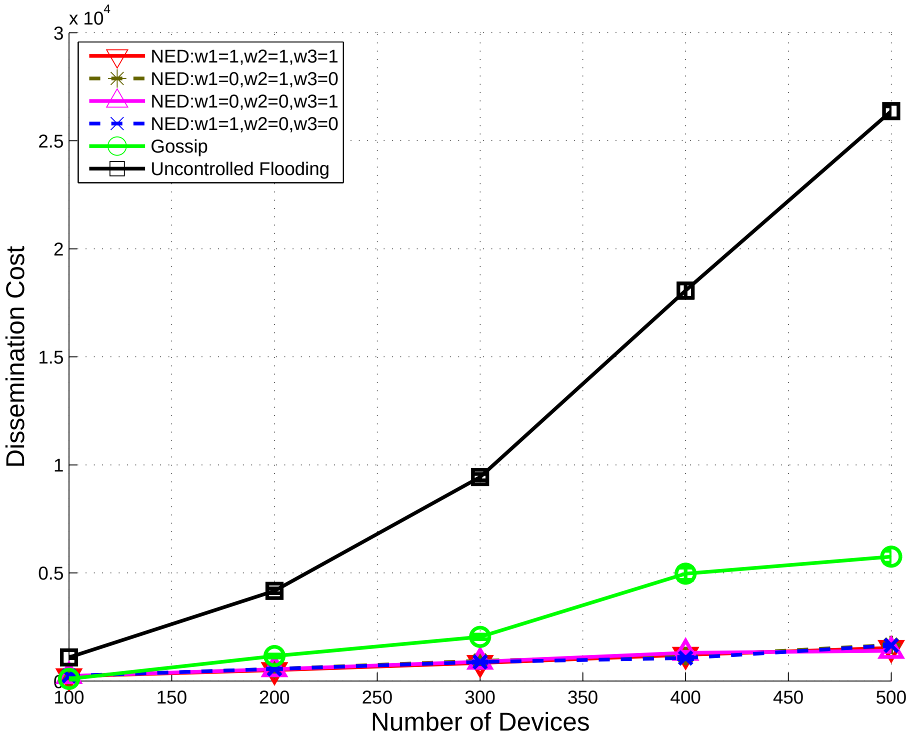

Figure B.4: Dissemination cost in random scenarios

The behavior of NED is explained by the cooperation between devices, according to a dynamic adjustment of the depths functionality, i.e., devices do not necessary need to reach the whole network until the maximum depth value. For example, when NED exhausts the possible paths to continue the dissemination, only the device that belongs to the NED path will disseminate the management information, contributing to an efficient dissemination of messages. On the other hand, Gossip performs controlled flooding, which introduces an extra cost proportional to the number of devices in the network. The Gossip behavior is explained by the probability function used to define how many neighbors should the information be flooded. This probability is calculated according to the number of devices directly connected, where the maximum number of neighbors to perform the flooding was defined to be 25% of the dissemination devices. Uncontrolled flooding has the highest number of messages, since all devices flood the management information.

The weights combination of NED does not have a significantly impact on the dissemination cost for both scenarios. In fact, for each repeated simulation, the bandwidth and local resources for each devices are regenerated without any pattern, reducing the influence of NED weights ($\boldsymbol{w1}$,$\boldsymbol{w2}$, $\boldsymbol{w3}$). The weight $\boldsymbol{w3}$ should have more influence in the dissemination cost, but due to the nearly uniform placement of the devices in both scenarios, it balances the network density value (Equation B.3) for all the device's neighbors.

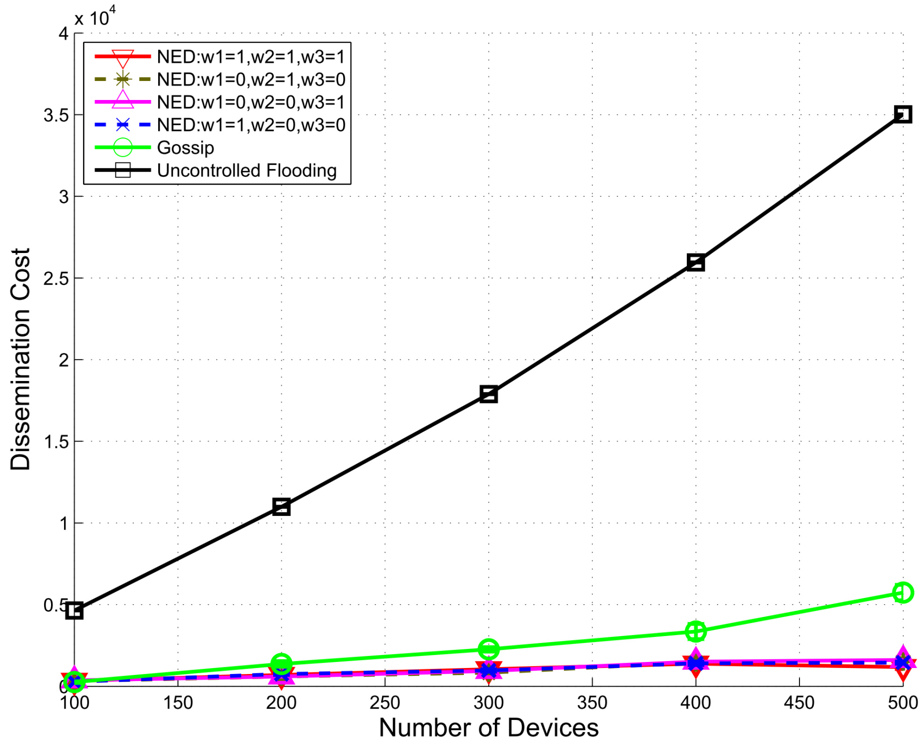

Figure B.5: Dissemination cost in realistic scenarios

B.4.4 Recursive Cycles

The analysis of recursive cycle, in Figures B.6 and B.7, is important to assess the impact of the three approaches in terms of resource consumption (e.g. memory usage and cpu processing). The lower is the number of recursive cycles, the lower is the consumption of the device resources.

The uncontrolled flooding has the highest number of cycles, since all devices flood the management information through the network. The NED reduces the recursive cycles when compared with Gossip, since in Gossip, part of the devices flood the network with management information, in both uniform random and realistic scenarios, while NED choses paths that prioritize bandwidth available and the devices' density, spreading more information with less depth between the dissemination devices. The increasing number of devices enlarges the difference of the number of recursive cycles of the three approaches, since it is almost constant in NED due to the dynamic adaptation of the depth, while in Gossip and uncontrolled flooding it grows.

B.4.5 Disseminated Devices

In random scenarios (Figure B.8) the number of disseminated devices is similar for the three approaches, except for 100 devices. For 100 devices, Gossip does not reach all the devices of the network, due to the 25% of the dissemination devices. NED is able to disseminate the management information about all devices of the network for a smaller dissemination cost and lower number of recursive cycles.

Only uncontrolled flooding reached the total number of disseminated devices in the realistic

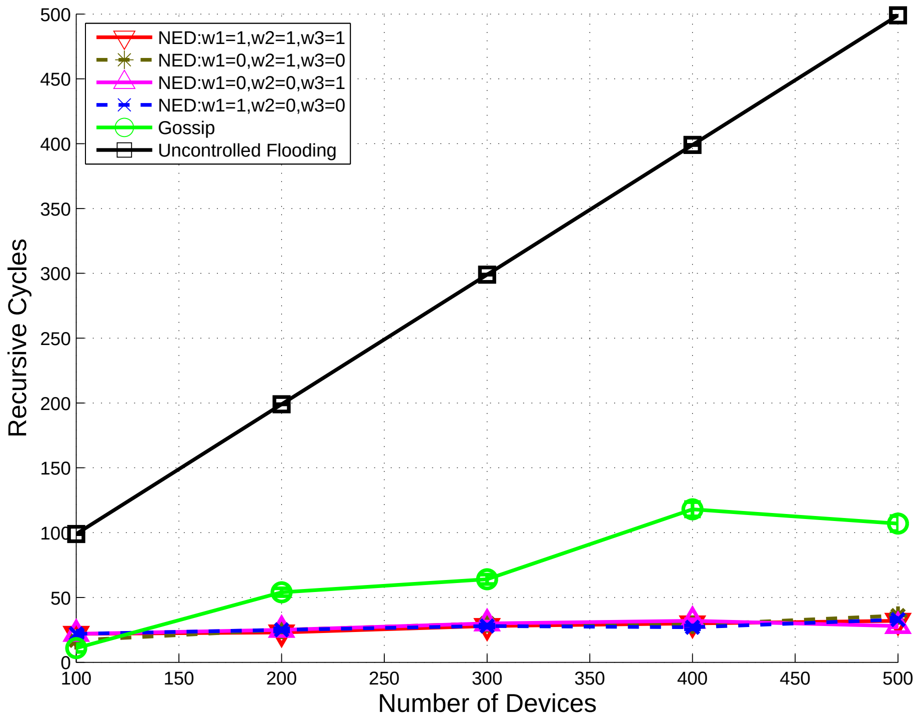

Figure B.6: Recursive cycles in random scenarios

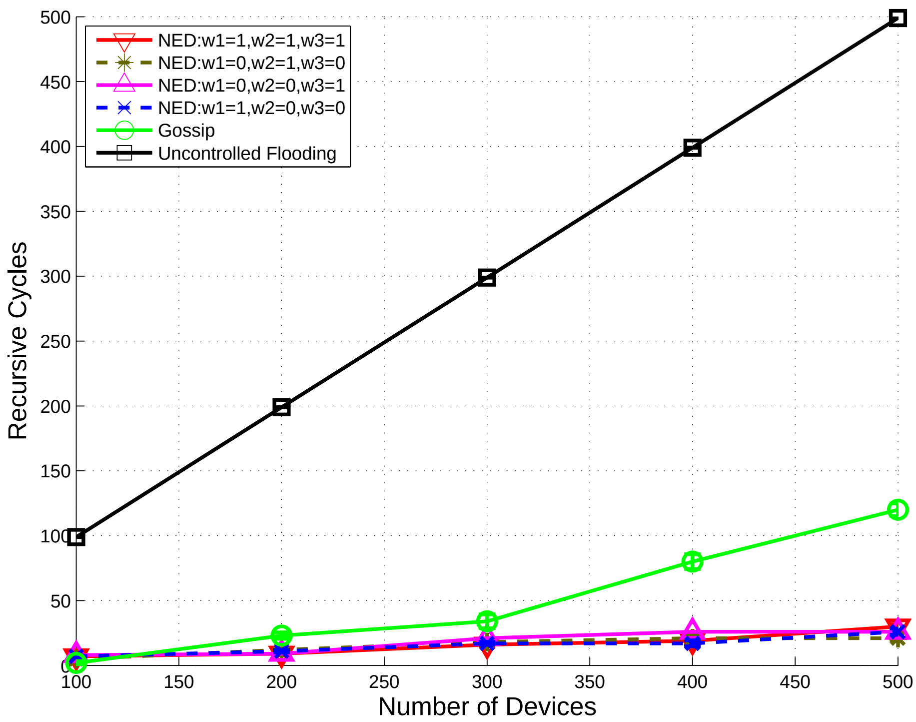

Figure B.7: Recursive cycles in realistic scenarios

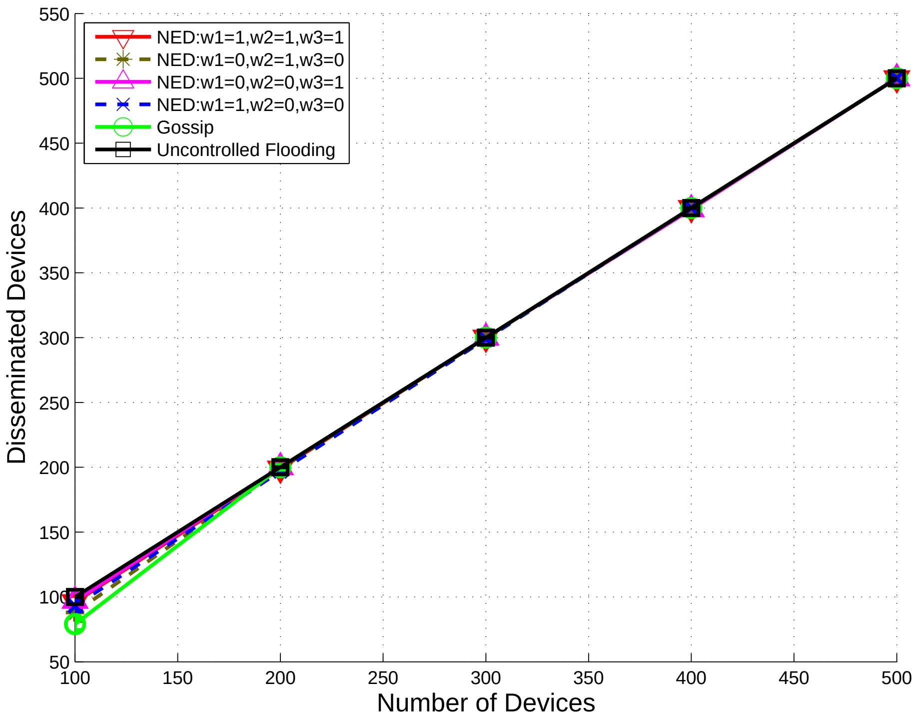

Figure B.8: Disseminated devices in random scenarios

scenario (Figure B.9). The Gossip depends on the amount of neighbor's devices that perform the dissemination of the management information (25% of dissemination devices), whereas NED depends on the dissemination devices to reach the total number of devices in the network. In this case, both 10% of dissemination devices and 100m of wireless range are not sufficient to reach all devices in the realistic scenario.

A trade-off between the number of dissemination devices and the disseminated devices is required, which should be achieved considering the wireless range of the devices and the considered area. Table B.1 extends the analysis of the three evaluated metrics for the three approaches, evaluating the impact of the number of dissemination devices (e.g. 20% and 80%), and the wireless range (e.g. 50m, 100m and 150m) for 500 devices in a realistic scenario. We consider the NED weights ($\boldsymbol{w1 = 1, w2 = 1, w3 = 1}$) in this evaluation.

In Gossip the increase of dissemination devices and consequently the increase of the maximum number of neighbors to perform the flooding dissemination strongly increases the dissemination cost and recursive cycles, especially for higher wireless ranges. In NED it reduces the dissemination cost and recursive cycles for higher wireless ranges.

From Table B.1 we obseve that NED with more dissemination devices decreases the dissemination cost and recursive cycles for higher wireless ranges (e.g. 100 and 150). For lower wireless ranges (e.g. 50), less dissemination devices decreases both dissemination cost and recursive cycles, but it does not disseminate the management information through all the devices of the network.

B.5 Conclusions and Future Work

This paper analyses the Neighbors Eyesight Direction (NED) dissemination for three different scenarios, wired, random and realistic devices placement. The NED proved to be

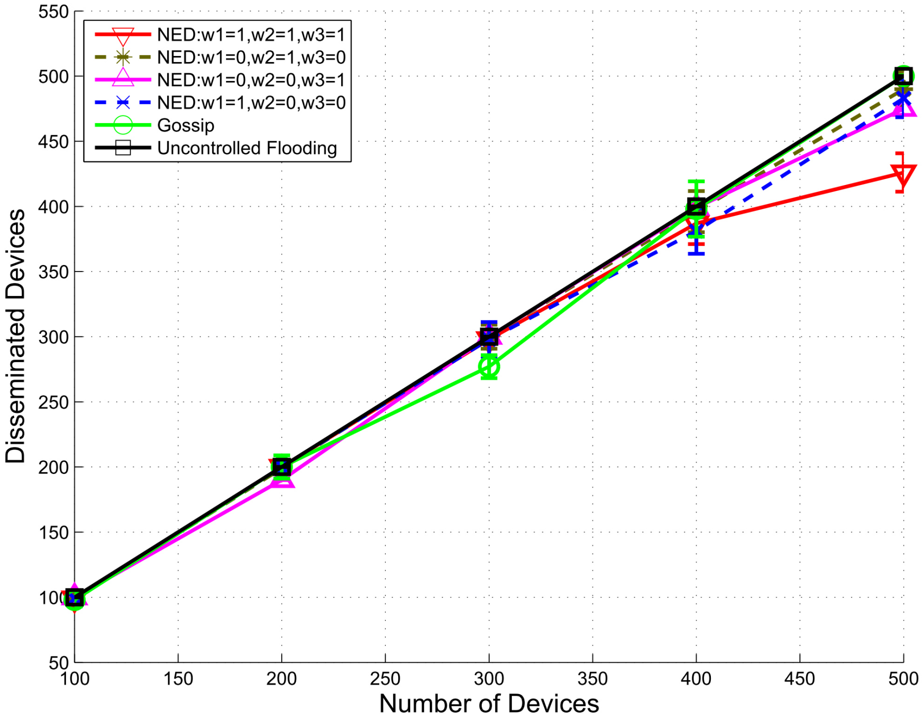

Figure B.9: Disseminated devices in realistic scenarios

Table B.1: Evaluation for realistic scenarios with 500 devices: 20% and 80% of dissemination devices; and 50m, 100m and 150m of wireless range

20% Dissemination Devices

NED	***Dissemination Cost***	***Recursive Cycles***	***Disseminated Devices***
50m	1294 ± 66.9619	77 ± 4.9660	450 ± 13.1572
100m	1905 ± 123.1851	29 ± 1.4001	500 ± 0
150m	1844 ± 74.1856	17 ± 0.6940	500 ± 0
Gossip	***Dissemination Cost***	***Recursive Cycles***	***Disseminated Devices***
50m	8495 ± 101.3967	476 ± 2.9209	500 ± 0
100m	11146 ± 376.1676	203 ± 6.5126	500 ± 0
150m	11990 ± 550.2752	105 ± 6.5126	500 ± 0

80% Dissemination Devices

NED	***Dissemination Cost***	***Recursive Cycles***	***Disseminated Devices***
50m	1682 ± 24.0675	98 ± 1.2180	500 ± 0.2594
100m	1885 ± 63.0196	28 ± 0.8139	500 ± 0
150m	1500 ± 93.3205	16 ± 0.4021	500 ± 0
Gossip	***Dissemination Cost***	***Recursive Cycles***	***Disseminated Devices***
50m	9823 ± 0	499 ± 0	500 ± 0
100m	35015 ± 0	499 ± 0	500 ± 0
150m	68549 ± 225.3114	489 ± 1.0637	500 ± 0

scalable for the evaluated scenarios, and also enforces the devices to search the best neighbor direction. It contributes to improve resource aggregations, reducing the dissemination cost and the device's resources consumption, in the dissemination of the management information for wired and dense wireless networks. The NED brings benefits to the distributed management process, due the possibility to control the amount of information to be disseminated in the network, by enabling cooperation between devices. As future work, we plan to adjust the NED weights in real time to optimize the number of discovered nodes or other relevant criteria, according to the network context.

B.6 Acknowledgments

This work is supported by the Fundação para Ciência e Tecnologia - FCT, through the grants SFRH/BD/62511/2009 and SFRH/BD/65265/2009.

References

[BM09] M. Bratislav and M. Miroslaw. Npart - node placement algorithm for realistic topologies in wireless multihop network simulation. In *SIMUTools*, 2009.

[CHLS07] C. Chi, D. Huang, D. Lee, and X. Sun. Lazy flooding: a new technique for information dissemination in distributed network systems. *IEEE/ACM Trans. Netw.*, 15(1):80–92, 2007.

[CMS08] C. Chen, J. Ma, and J. Salomaa. Simulation study of cluster based data dissemination for wireless sensor networks with mobile sinks. In *Proc. 10th International Conference on Advanced Communication Technology ICACT 2008*, volume 1, pages 231–236, 17–20 Feb. 2008.

[DBS+07] S. Davy, K. Barrett, M. Serrano, J. Strassner, B. Jennings, and S. Van der Meer. Policy interactions and management of traffic engineering services based on ontologies. In *Network Operations and Management Symposium, 2007. LANOMS 2007. Latin American*, pages 95–105, 2007.

[DDL+02] N. C. Damianou, N. Dulay, E. C. Lupu, M. S. Sloman, and T. Tonouchi. Tools for domain-based policy management of distributed systems. In *In Proceedings of the IEEE/IFIP Network Operations and Management Symposium (NOMS 2002)*, pages 203–217, 2002.

[DFKS07] V. Drabkin, R. Friedman, G. Kliot, and M. Segal. Rapid: Reliable probabilistic dissemination in wireless ad-hoc networks, proc. 26th ieee srds. pages 13–22, 10–12 Oct. 2007.

[Gea] L. Guardalben and et al. Lightweight discovery and exchange of network information in distributed network management. Ghent, Belgium, Integrated Management IM, May 2013.

[Gen07] *EraMobile: Epidemic-Based Reliable and Adaptive Multicast for MANETs*, March 2007, WCNC 2007.

[GGP+11] L. Guardalben, T. Gomes, A. Pinho, P. Salvador, and S. Sargento. Nodes discovery in the in-network management communication framework. In *MONAMI*, pages 145–157, 2011.

[Mat05] Matlab central. *Probabilistic Flooding Simulation*, 2005. http://www.mathworks.com/matlabcentral/fileexchange/7928-probabilistic-flooding-simulation.

[MAT11] MATLAB. *version 7.13.0 (R2011b).* The MathWorks Inc., 2011. Available at `http://www.mathworks.com/products/matlab/`.

[Meg06] N. Meghanathan. An algorithm to determine the sequence of stable connected dominating sets in mobile ad hoc networks,proc. aict-iciw '06. pages 32–32, 19–25 Feb. 2006.

[ML08] e. a. Mingzhe Li. Wbest: a bandwidth estimation tool for ieee 802.11 wireless networks. In *LCN, Montreal, Quebec, Canada*, 2008.

[Moy98] J. Moy. *OSPF Version 2.* IETF RFC 2328, April 1998. Available at `http://www.ietf.org/rfc/rfc2328.txt`.

[SCM05] H. Sabbineni, K. Chakrabarty, and S. Member. Location-aided flooding: An energy-efficient data dissemination protocol for wireless sensor networks. *IEEE Transactions on Computers*, 54:36–46, 2005.

[ST05] M. Sbodio and W. Thronicke. Ontology-based context management components for service oriented architectures on wearable devices. In *Industrial Informatics, 2005. INDIN '05. 2005 3rd IEEE International Conference on*, pages 129–133, 2005.

[Tan02] A. Tanenbaum. *Computer Networks book.* Prentice Hall Professional Technical Reference, 4th edition, 2002.

[Zhi09] Z. Zhizhao. Distribution policy based information dissemination management system. volume 4, pages 325–328, Information Management, Innovation Management and Industrial Engineering, 2009 International Conference on, 2009.

Annex C

Improving MAC Layer Association through Social-Based Metrics in Mobile Networks

Lucas Guardalben, Tomé Gomes, Paulo Salvador and Susana Sargento

In IEEE Communication Magazine, June 2012

The format has been revised.

Improving MAC Layer Association through Social-Based Metrics in Mobile Networks

Lucas Guardalben, Tomé Gomes, Paulo Salvador and Susana Sargento

Abstract

Mobile Social Network (MSN) is a social network where one or more elements share interests or commonalities by connecting with one another using mobile devices. However, contrary to traditional online social networks, efficient communication and information exchanged depends on mobile users' context. In order to increase the scalability of the information exchange, network nodes need to be associated and grouped through an efficient set of rules and criterions, to minimize the time and amount of data messages flowing in the network. This article shows how social-based metrics can increase the efficiency of layer 2 associations in mobile networks, where nodes are able to join the network through their context information and probable interactions, optimizing the information exchange at MAC layer. Through a case study of mobile communications with different mobility patterns and dissemination approaches, we show that the social-based metrics increase overall network efficiency: maximize the percentage of information delivery, decrease the time to disseminate data traffic and reduce network usage.

C.1 Introduction

The enormous growth of mobile networks opens opportunities for a synergy among mobile applications and social networks, leading the users to extend the possibility to create a Mobile Social Network (MSN) [Lug08] anywhere at anytime. Social networks allow the user to search other users with mutual interests, establishing and maintaining social communication infrastructures with each other, using mobile devices such as laptops, smartphones and tablets. The mobile social networks need to be able to adapt autonomously to mobile environment changes, according to context informations and mobility pattern of their users. Additionally, new generation of mobile devices provide sensing capabilities (e.g. GPS, Cameras, Accelerometers, etc.), which enable the users to interact with each other directly through many ways. This is an advantage when the user looks forward to a more interactive environment. Nevertheless, this high degree of dynamism and interactivity requires that the mobile devices use lightweight mechanisms, for example, from simple initial exchange of information, or to share multimedia content.

In mobile social networking users can also access virtual communities and share their social interests between themselves. Recent studies [NDXT11, PP09] have shown how the social communities are formed and grouped by interests providing helpful information in order to develop social-aware strategies for social networks problems. These concepts may inspire the understanding of, for example, how wireless communities can be structured so that the users can share their personal interests. However, mobile social networks are limited by restricted functionalities of mobile devices which provide various challenges at different levels [QMK10]. Frequent disconnections due to power battery constraints, poor signal/bandwidth, lack of specialized infrastructure and poor scalability of exchanged information, are examples of these limitations. Generally, mobile social networks are created using existing infrastructures such as IEEE 802.11 Mobile Ad-hoc Networks (MANETs) [IEE07], when inherit these known problems, thoroughly investigated by the scientific community.

To overcome these limitations, it is desired to provide mechanisms that work in the MAC layer, ensuring a lightweight communication process due to the high amount of information exchanged between the nodes. In fact, working at MAC layer ensures a more efficient and scalable information-oriented service at upper layers. In the IEEE 802.11, both in the Basic Service Set (BSS) and Independent BSS (IBSS), the criteria for nodes' association is usually the Received Signal Strength Indication (RSSI) and Signal to Noise Ratio (SNR). However, having higher RSSI and SNR in some cases does not mean higher throughput, since these values depend on the distance between the transmitter and receiver, as well as on the transmission power (TX/RX) [AKET09]. Moreover, this association criteria was proved to be inefficient in terms of load balancing [LMY+11] and network performance [AKET09].

This article evaluates how social-based association models, where the nodes are grouped in communities through social metrics, can improve communication stability and efficiency. By the definition of new social metrics, this article presents a mechanism to group nodes into communities according to social interactions with the aim to improve their message exchange for the support of efficient dissemination of information in wireless networks. These social metrics measure and predict the friendship of neighborhood nodes, associated nodes and community of nodes, so that the connection between nodes can be established with the objective to optimize the network usage and efficiency of data dissemination. Additionally, the understanding of friendship is quantified by the relation/interaction between the nodes at MAC layer in the social mobile network. The implementation of a social-based association model in a set of use case scenarios showed that the social-based metrics maximize the percentage of information delivery, decrease the time to disseminate data traffic, and reduce network usage in mobile social network scenarios.

The remainder of the article is organized as follows. Section C.2 presents social network approaches in delay tolerant networks, which inspire the support of social approaches in the usual mobile communication scenarios. Section C.3 introduces the social-based association model on mobile social communication infrastructure. Section C.4 describes the social-based metrics, and Section C.5 presents the obtained results in several case-studies. Finally, Section C.6 concludes the article.

C.2 Social Network Inspirations: Related Work

The solutions inspired by social networks are mainly found in the field of routing for Delay Tolerant Networks (DTNs) [MMDA10, DH07], where the main idea is to use metrics to predict the movement of nodes based on the repetition of nodes contacts. The problem of choosing the best forwarders in DTNs is shown in [FV11], where the authors introduce sociability in routing, a novel routing scheme that selects a subset of optimal forwarders among all the nodes and relies on them for an efficient delivery. However, these metrics take into account only information at network layer, capturing only frequency and type of encounters of nodes. The work proposed in [LDS03] makes use of the observations of the non-randomness of mobility and, to improve routing performance, the authors propose probabilistic routing, PROPHET, which uses history of encounters and transitivity. Even so, it is not guaranteed that a node with a higher metric will be encountered within reasonable time. In [HCY11] the authors seek to understand the human mobility in terms of social structures. The authors also emphasize that it is possible to detect characteristic properties of social grouping in a decentralized manner from a diverse set of real world traces. However, this grouping work with a hierarchical community structure which tend to adapt slowly to changing needs. A novel technique determining the impact of human mobility in the design of forwarding opportunities protocols on delay tolerant networks is proposed in [HCY11]. With regard to provide social mobile network through MANET's as infrastructure base, very few efforts have been done [MH11]. In [DH07] the authors presented a social network analysis for routing in delay tolerant MANET's. However, this approach is limited on the use of already existing routing protocols for MANET's.

C.3 Bringing Social-Based Metrics at MAC Layer

The central idea of bringing social network concepts at MAC layer is to improve the overall data traffic performance, from the association process to the dissemination of data traffic. The social-based association model is illustrated in Figure C.1. It is composed by cooperative mechanisms plus the concepts of social metrics (described in section C.4) at MAC Layer in mobile social communication infrastructure. The social network concepts are incorporated through social-based metrics in the association process, replacing the default criteria such as RSSI-based or first received Beacon. In the case of IEEE 802.11 networks, social-based metrics work at IEEE 802.11 MAC layer, and the extensions at signaling and association processes remain compliant to the IEEE 802.11 standard.

C.3.1 Social-Based Association Model

The social-based association model is embedded in each node within the mobile social communication infrastructure, which consists of two main blocks: local repositories and co-operation mechanisms. The first one stores the information gathered from received Beacons

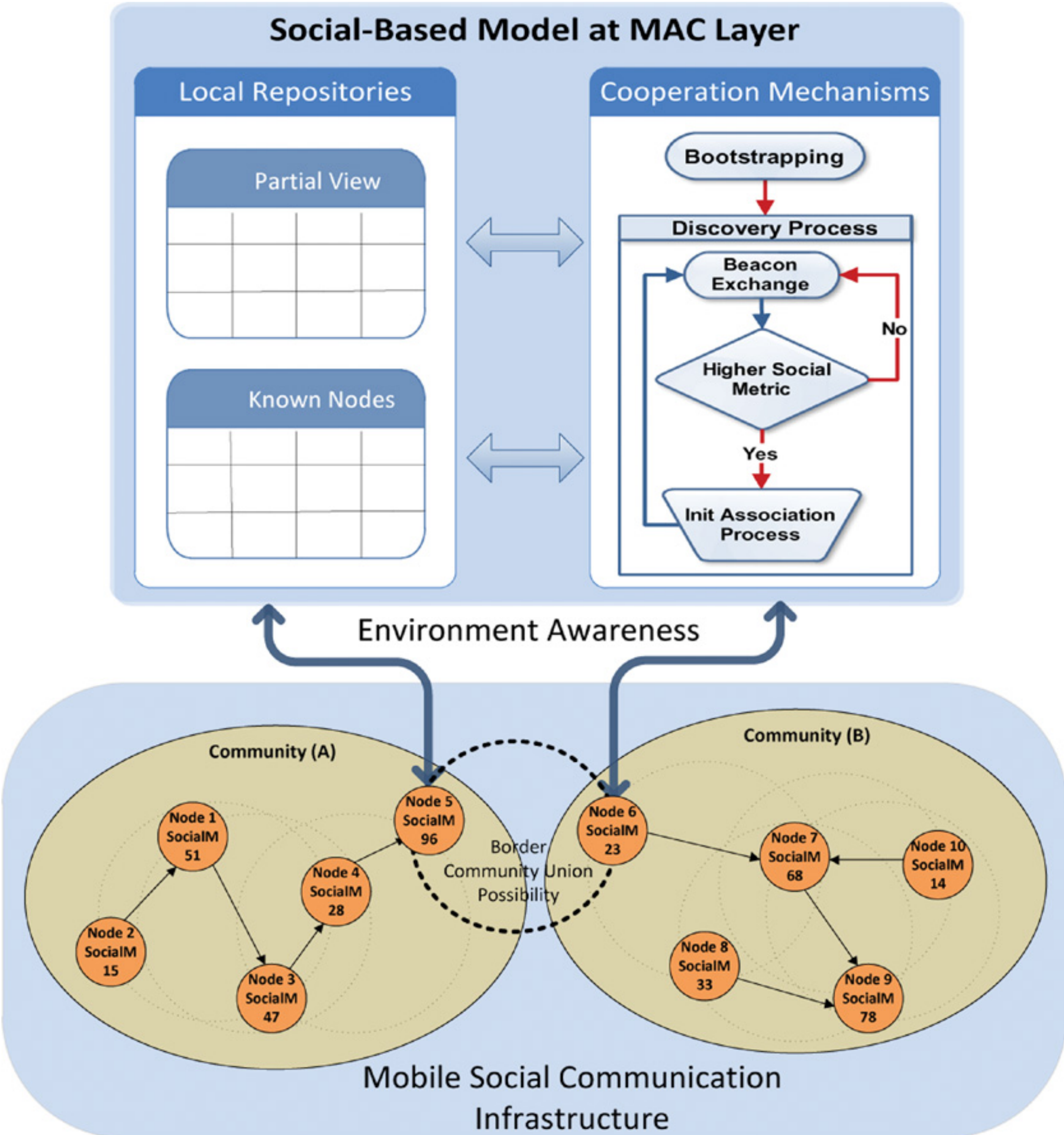

Figure C.1: Social-Based Association Model

into local tables: Partial View, which contains information of nodes in the neighborhood; and the Known Nodes, which contains information about nodes found through other nodes in a specified range. The cooperative model incorporates the mechanisms of Bootstrapping and Discovery. These mechanisms of cooperation in distributed architectures aim to increase the efficiency and speed in the convergence of information. This information relates mainly to the nodes which belong to a community and their characteristics.

The model needs to incorporate the high degree of dynamism of the scenarios, where nodes can join or leave a community at any time. Through these mechanisms, the relevant information is collected from the network, which is then inserted in the local repositories in order to constantly improve and update the knowledge about the other nodes. This knowledge acquisition through collaboration between the nodes is necessary but complex, due to the lack of common points for synchronization. Therefore, the following mechanisms are present in the cooperation approach.

C.3.1.1 Bootstrapping

From a single node viewpoint, the bootstrapping will set up the initial information, e.g. nodes identifiers, initial management policy strategies, assignment of unique identifier, MAC address, BSSID, hardware capabilities. At this stage the node is ready to start its activity on the network. Then, the bootstrapping will call the discovery mechanism, enabling new users to join the network communication as well.

C.3.1.2 Discovery

A node detects and identifies other nodes in the network to create new links of association. In this process each node becomes aware of neighbors presence by locally exchanging information among them. The discovery can be made by direct or indirect contacts (e.g between 1-hop or multi-hop nodes). The mechanism used for this purpose is the exchange of IEEE 802.11 MAC packets (e.g Beacons, Associations and Reassociations) with additional Information Elements (IE's) [IEE07] as referred at Subsection C.3.2, allowing for example, the association to a node with higher social metric. Note that the discovery process refers to the continuous process of maintaining the information updated through the Beacon exchange process. After the beaconing process and all initial information is gathered, the nodes are able to decide which node in their vicinity is the best association regarding the social metric value. The nodes are able to change their association link according to the current environment conditions.

C.3.1.3 Association Process

The process of association is made to the neighbor node whose social metric is higher. Therefore, communities are created and optimized in a distributed way. In parallel, it takes place the process of maintaining and propagating the community consistence, since associations are not definitive and nodes can join in different communities. Regarding the communities infrastructure side, the social-based metric enables the communities to split or join according to nodes positions and interactions. Having a network divided into self-organized communities is advantageous since each node does not need to have an overview of the entire network, but only of the community where that node is, reducing the amount of information needed to collect and store. Assuming that a community has a parent-child relation structure, where the node that initiated an association is the child of the other node (i.e. the parent). A distributed algorithm was developed in order to estimate the communities size. It uses a distributed backtracking process, where: (i) all nodes report to their parent node the number of nodes associated to them, (ii) the head node (i.e. the first node of the community and overall parent) process all reports of its child nodes, estimates the size of the community and announces back the result to all its child nodes and (iii), subsequently, all nodes report to all their child nodes the size of the community announced by the head node.

C.3.2 IEEE 802.11 MAC Layer proposed extensions

Considering the IEEE 802.11 [IEE07] as the base standard and protocol, it was necessary to extend it to meet the requirements of the proposed mechanisms and social-based metrics. At the level of management frame Beacon and Association Response, the introduction of new

Information Elements (see Figure C.2 (a) and (b)) in the Frame Body field allows the nodes to cooperate in order to share information between them (e.g. fill the local tables, Partial View and Known Nodes, with the knowledge of each node).

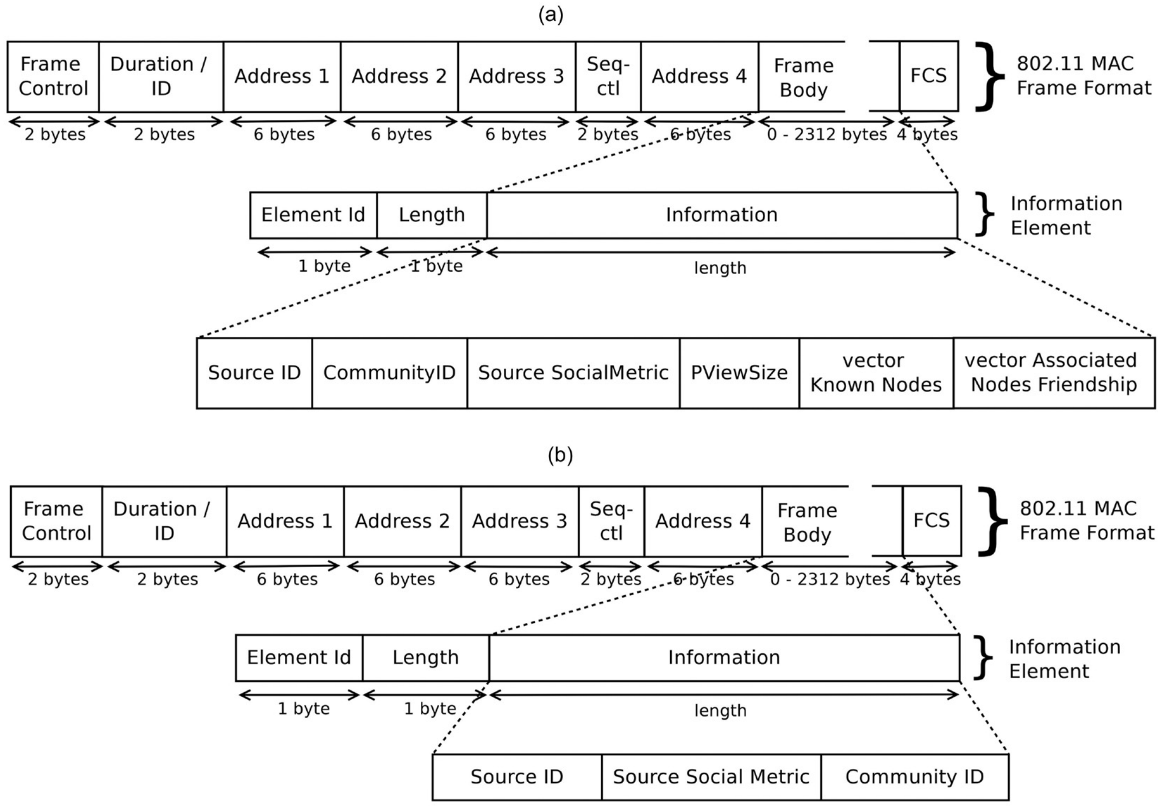

Figure C.2: (a): Community-Based Beacon IE - (b): Community-Based Association Response IE

To perform the decisions at local level, base on social-based metrics a new community-based Association Information Element (see Figure C.2 (b)) is required. With regard to Information Element fields, the *Source ID*, *CommunityID* and *SocialMetric* fields identify the node and community which the node belongs to, and its current social metric value; *PViewSize* refers to the number of neighbors that each node reports in the exchanged management packets, and *RecBeacons* is the number of Beacons received from each neighbor since it entered in the vicinity. The next section will define the social-based metrics.

C.4 Social-Metrics at MAC Layer

As previously referred, we consider that the association between nodes is no longer made through RSSI-Based approaches: the association is performed to the node with the highest social metric in the vicinity. The main idea is to use the knowledge obtained from interactions of each node and its surrounding neighbors, through the concepts of communities, density/quality of the friends and 'friendship' of nodes. 'Friendship' of nodes refers to how close the nodes are and their respective quality (e.g local and network resources), and also the amount of nodes known by those nodes, inside or not of a community. Three complementary social-aware metrics were proposed: (i) *Neighborhood Nodes Friendship* (ii) *Associated Nodes Friendship* and (iii) *Community Nodes Friendship*.

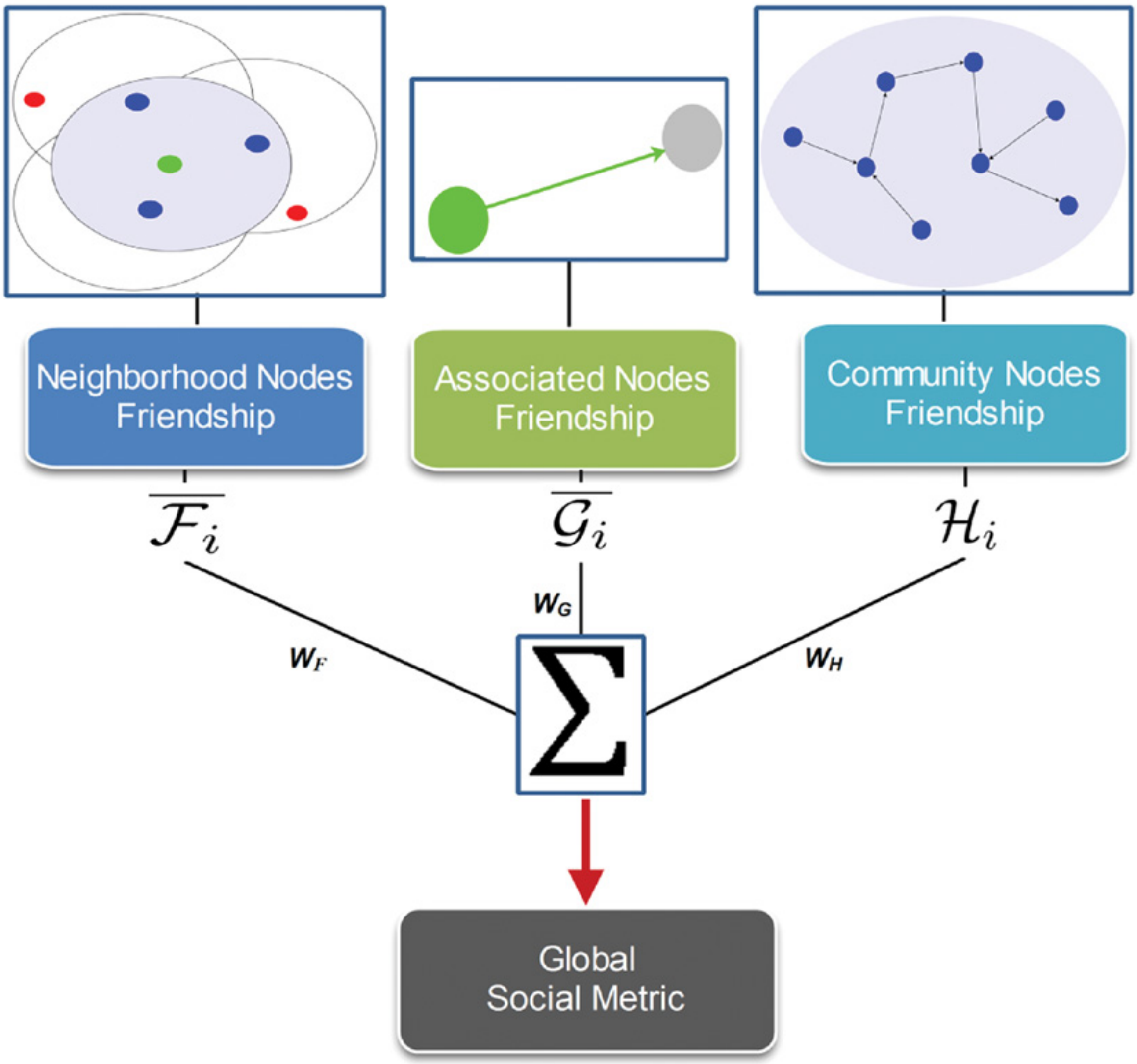

Figure C.3: Graphical Representation of the Social Metric

The model assumes a distributed mobile wireless network consisting of nodes that interact with each other through direct or indirect contacts, forming a fully connected network in the coverage area, where nodes have at least one common channel between their neighbors.

We define $\mathscr{N}$ as the complete set of nodes in the scenario: $\mathscr{N} = \{n_i : i = 1, \ldots, N\}$. Through the discovery process, each node knows, at least, the set of nodes in the neighborhood that can be defined by: $\mathscr{V}_i = \{n_j : n_j \in \mathscr{N} \wedge j \neq i \wedge v(i,j) = 1\}$, where $v(i,j)$ is a function that returns 1 if nodes i and j are neighbors and 0 otherwise. The (sub)set of nodes associated with node i can be defined by: $\mathscr{A}_i = \{n_j : n_j \in \mathscr{N} \wedge j \neq i \wedge a(i,j) = 1\}$, where $a(i,j)$ is a function that returns 1 if nodes i and j are associated and 0 otherwise. Note that, $\forall i, \mathscr{A}_i \subseteq \mathscr{V}_i \subseteq \mathscr{N}$. A community is a group of nodes directly associated or with a multi-hop path between them. The subset of nodes in the same community of node i is defined by: $\mathscr{C}_i = \{n_j : n_j \in \mathscr{N} \wedge c(j) = c(i)\}$, where $c(i)$ is a function that returns the community identifier where node i belongs. Note that, $\forall i, \mathscr{C}_i \subseteq \mathscr{N}$ and $\forall i, |\mathscr{A}_i| \leq |\mathscr{C}_i| \leq |\mathscr{N}|$, where $|.|$ represents the cardinality of a set. Finally, the metrics described in the following sections will be jointly grouped in a social-based association metric as presented in the Figure C.3.

C.4.1 Neighborhood Nodes Friendship

The purpose of this metric is to quantify the degree of 'friendship' between any node and its surrounding neighbors. This 'friendship' will evaluate the number of neighbors and the relative degree of interaction between them in terms of number of exchanged Beacons. Therefore, it evaluates the surrounding conditions independently of the association chosen by nodes, giving it a perspective of potential expansion of the neighborhood. Note that, in a mobile wireless scenario, these conditions change over time, so the node must monitor them constantly. The result measures the relevance that each neighbor has in terms of direct communication and 2-hop density of nodes, and its value is an average normalized between $[0,1]$.

The overall 'friendship' of a node i can then be inferred averaging the 'friendship' between node i and all other nodes in its neighborhood ($\mathscr{V}_i$):

$$\overline{\mathcal{F}_i} = \frac{1}{|\mathscr{V}_i|} \sum_{j:n_j \in \mathscr{V}_i} \left(\frac{b_{i,j}}{\sum_{k:n_k \in \mathscr{V}_i} b_{i,k}} \times \frac{|\mathscr{V}_j| + 1}{|\mathscr{V}_i| + 1} \right), \tag{C.1}$$

where $b_{i,j}$ represents the number of beacons received by node i from a node j, and $|\mathcal{V}_i|$ represents the cardinality of the set of neighbors of node i, i.e., the size of the neighborhood of node i plus itself.

C.4.2 Associated Nodes Friendship

The intention of this metric is to quantify the quality of 'friendship' of a node associated to a neighbor node, measuring the quality of the association between them. This is a one-to-one measure, where it is a proportion between the SNR over the association link, and the stability of this link according to the Beacons exchange since the association, and the level of local resources. The result is an average multiplication of these three features normalized between $[0,1]$, allowing a constant monitoring of the chosen association link.

The main concept here is to constantly know how is the chosen link for association, both in terms of the surroundings and the local capabilities of the node, and is given by:

$$\overline{\mathcal{G}_i} = \frac{1}{|\mathscr{A}_i|} \sum_{j:n_j \in \mathscr{A}_i} \left(\rho_{i,j} \times \frac{b_{i,j}}{s_j} \times \mathcal{R}_j \right), \tag{C.2}$$

where $\rho_{i,j}$ is an estimation of the Signal to Noise Ratio between nodes i and j; $\frac{b_{i,j}}{s_j}$ is an estimation of the link stability between the nodes (ratio between the number of Beacons sent by node j and those that were actually received by node i ($b_{i,j}$), and the total number of beacons sent by node j (s_j)); and $\mathcal{R}_j$ is a metric that quantifies node j available resources characteristics (such as processor, memory, storage, battery remaining and type of interfaces).

C.4.3 Community Nodes Friendship

This metric quantifies the quality of 'friendship' between a node and the nodes in the same community. The idea is to combine the size of the community (given by the distributed algorithm explained in the subsection C.3.1.3) together with the average overall 'friendship' quality of all nodes in a community.

$$\mathcal{H}_i = \sum_{j:j \in \mathscr{C}_i} \overline{\mathcal{G}_j}, \tag{C.3}$$

where $|\mathscr{C}_i|$ represents the cardinality of the set of nodes in the same community as node $\boldsymbol{i}$, i.e., the number of nodes in the same community of node $\boldsymbol{i}$.

C.4.4 Global Social Metric

The global social metric is obtained by performing a weighted sum of the three complementary metrics (neighborhood, associated and community), defining $\boldsymbol{w_F}, \boldsymbol{w_G}$ and $\boldsymbol{w_H}$ as the weights of the metrics, respectively. It is possible to define the global social metric as referred in Figure C.3, where the sum of weights is 1. In addition, the weights can be set up deterministically, or by experimental combinations to chose the best combination.

C.5 A Social-Based Metrics: Case studies in Mobile Scenarios

This section evaluates the social-based metrics in case studies of IEEE 802.11 networks with mobile nodes with different mobility patterns. In our scenarios we assume a distributed wireless network consisting of mobile nodes which communicate through peer-to-peer interactions within the limited coverage of each other. The social-based association model is developed in NS-3 simulation environment [NS-10], considering three mobility patterns: *Random Walk*, *Random Waypoint* and *Nomadic*.

With regard to mobile patterns, *Random Walk* produces random mobility traces without any pattern, thus, conditions will vary quickly causing associations to be made and removed often. On the other hand, *Random Waypoint* causes nodes to travel to specific areas around an interest point with attraction capabilities (e.g points of interest in cities). Finally, *Nomadic* induces nodes to choose positions within an area of a mobile reference point, causing nodes to travel aggregated, similarly to a guided tour (e.g guided tour in museums, etc).

Additionally, the spread of information is performed by assigning a random node to start sending packets; some nodes will receive and forward them according to the dissemination mode, considering two different modes to disseminate the data information: *Pure broadcast* and *Hop-by-hop*.

In *Pure broadcast*, the nodes broadcast all the received data packets without any criteria and without the need for an established association process. This is approach used as a comparison baseline reference. On the other hand, *Hop-by-hop* creates an association path for dissemination, i.e., each node transmits the information by *unicast*, being the destination identified by the node where the association link has been established. In addition, we distinguish two types of association: *First Beacon* and *Social Metrics*. The first one is the default association mode of 802.11 MAC standard, where the criterion used is to establish the association link to the node from where the first Beacon came. The second is the social-based association model, where the criteria used is to establish the association link to the node with the higher social metric, and to constantly monitor the vicinity searching for higher social metrics. Additionally, both association modes were implemented in the 802.11 MAC layer level.

The set-up scenario is configurable, and those are the main settings:

- 100 nodes in a simulation window of 1000m x 1000m;
- Interval between dissemination flows exponentially distributed with average 1 second;
- *Random walk*, *Random Waypoint* and *Nomadic* mobility patterns with similar settings (e.g. speed of the nodes between [5, 10] m/s);
- *Pure broadcast* and *hop-by-hop* dissemination modes;

- *First Beacon* and *Social Metrics* association modes;
- The best weights $\boldsymbol{w_F}, \boldsymbol{w_G}$ and $\boldsymbol{w_H}$ for the social metric results is determined by the experimental combinations;
- Mechanism for loops detection and removal;
- The obtained results include the mean values of the results from independent repetitions with confidence intervals of 95%.

C.5.1 Percentage of Information Disseminated

Figure C.4 shows the percentage of nodes to which the dissemination information was received. Regarding to the *Random Walk* scenario, we observe a low percentage in all three association and dissemination modes. This is due to the fact that the nodes are all apart from each other and do not converge to common point(s), i.e., the movement between several nodes is independent and random in terms of direction and speed. In the other two mobility patterns (*Random Walk* and *Nomadic*), and considering the social metrics association mode, the percentages are higher in the scenario with fixed reference points (*Random Waypoint*) rather than those where the reference points are mobile (*Nomadic*). In a scenario where the convergence points are dynamic, the links between the several nodes are harder to maintain than in a scenario where those points are constant, therefore implying lower dissemination percentage achieved. Hence, the association links must be established accurately rather that randomly, and this is the role played by the social metrics, abstracting the concept of the best association and allowing a measurable evaluation of the current conditions of the terminal in social terms.

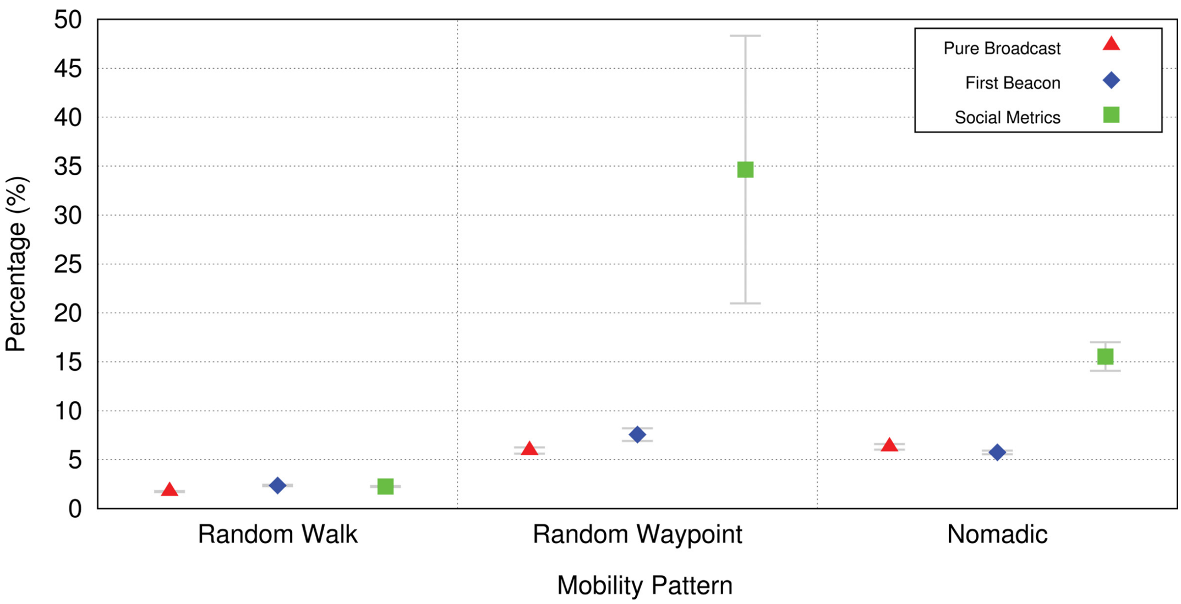

Figure C.4: Percentage of nodes that received the information disseminated

At a first glance, it would be expected that the *Pure Broadcast* dissemination mode reached, at least, the same percentage as the other two methods. However, we must keep in mind that these simulations were performed with 100 nodes, all sharing the same medium, being the access monitored by the Distributed Coordination Function (DCF). When a broadcast dissemination method is used, it is likely that more collisions occur and that the buffers fill more quickly, causing packet dropping. All these effects will have a negative impact on the results achieved. Unlike the *Pure Broadcast*, the other two approaches (*First Beacon* and *Social Metrics*) use the *hop-by-hop* as dissemination mode to forward data packets flow, thus becoming more efficient and achieving higher percentages. The difference between these last two is the criterion for association, which has influence in the final result, as we can see in the Figure C.4.

C.5.2 Time to Dissemination

In order to obtain a comparative analysis of the dissemination speed of the different modes in each scenario, it was chosen the reference percentage value of 5% from Figure 4 because it is the maximum (percentual) threshold achieved by the methods without social metrics. Thus, it was performed the analysis of the time that each mode takes to achieve that same reference. Also note that, with the *Random Walk* mobility, this percentage reference is not reached, hence this scenario is not shown in the Figure C.5.

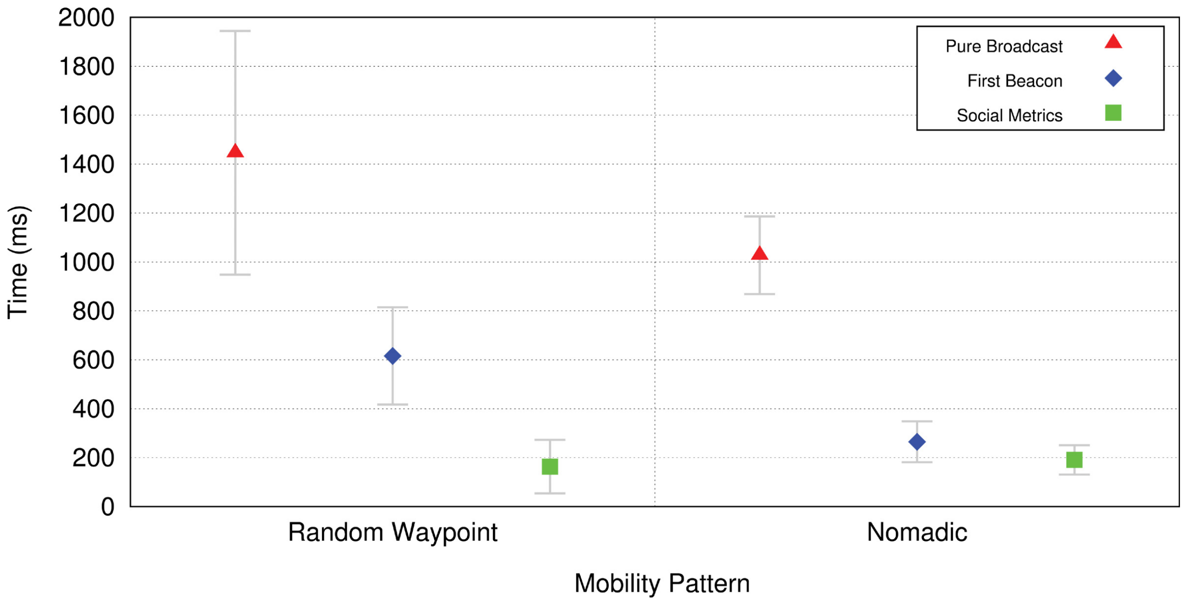

Figure C.5: Time for 5% of the nodes receive the information disseminated

Here we can see that, for the same percentage of nodes using the *Social Metrics* mode, it is possible to make a faster dissemination compared to the other two methods. This is due to the fact that, in this mode the association, links are established to the best neighbors according to the social assumptions (see Figure C.3), causing a more efficient *hop-by-hop* dissemination when compared to the mode *First Beacon*. Regarding to the *Pure Broadcast* mode, we can see that it takes a long time to reach the percentage reference. This is motivated by the reasons discussed previously for the low percentage, also reinforced by the fact that if all the nodes share the same medium, and if all nodes Broadcast all received packets, the time to gain access to the shared medium will be longer compared to the other two dissemination modes that use the association links for dissemination.

C.5.3 Network Usage

The strategy used to measure the network usage is to verify, in each moment, how many dissemination flows are active in the network, considering the interval between each one to be exponentially distributed with average of one second.

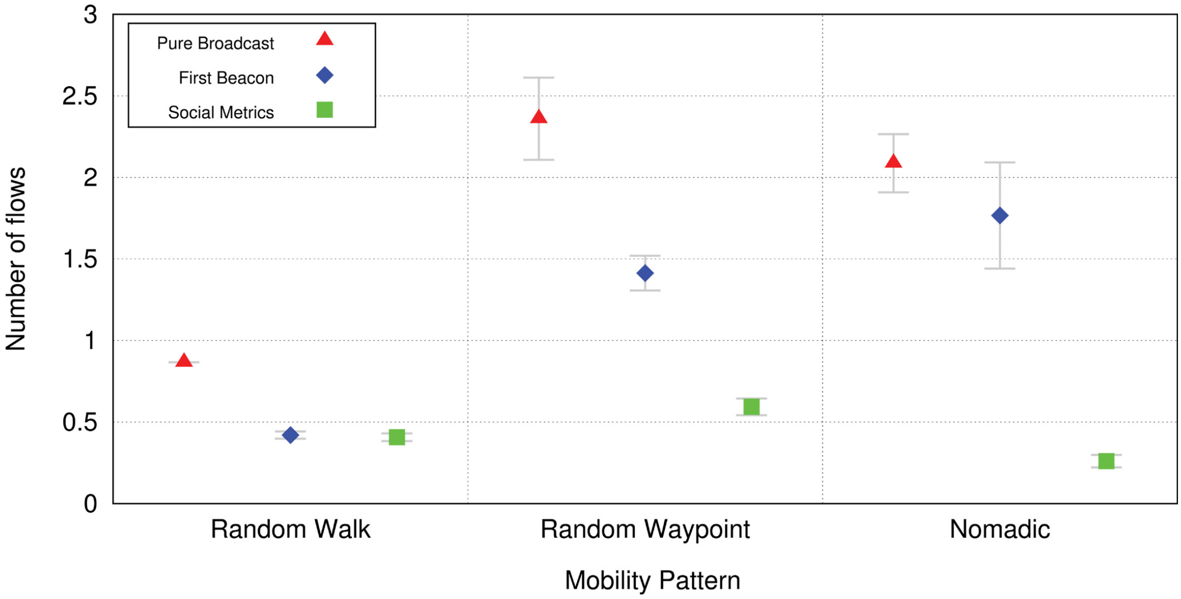

Figure C.6: Average Network Usage

In the Figure C.6 we observe that, with the *Random Walk*, the network usage is lower due, once more, to the lower percentage of information achieved with this mobility pattern (see Figure C.4). In the other two mobility pattern, the *Pure Broadcast* mode is, as expected, the dissemination mode that highly uses the network, i.e., with more overlapping flows that tend to remain in the network due to its uncontrolled broadcast nature. Comparing the other two dissemination modes, when the association is not permanent to the first node known but rather constantly improved according to our social criteria (see Figure C.3), the network usage can be significantly reduced, i.e., the efficiency of the dissemination mode can be maximized.

These results show the benefits provided by the social-based metrics in dissemination mechanisms, either in terms of percentage of nodes where this information is received, time to achieve a pre-defined percentage or network usage (i.e., dissemination efficiency and number of flows required in the network). Notice that this is extremely important to reduce the required data information exchanged between nodes. Moreover, the use of social metrics approach is extremely useful for scenarios where the high degree of mobility is the main requirement. The social metrics enforce the nodes to associate to the best neighbor nodes, contributing to improve the overall network communication process and increasingly the percentage of information dissemination, reducing the network time and utilization. This enforces a higher efficiency in the distributed mobile social networks scenarios.

C.6 Conclusion

This article presents a novel association model for wireless networks, where nodes are self-organized in communities based on social metrics criteria. We have shown that the usage of social-based metrics at MAC Layer outperforms the traditional technical association criteria.

The more positive impacts of bringing social-metrics to the lower network layers were observed in wireless network scenarios where terminals follow common mobility patterns (e.g. fixed waypoints reference and mobile guide tours). This approach is very promising since it will enable the support for efficient information delivery, reducing network usage and reducing information dissemination time. The main future challenge will be incorporating the social-based association model into an experimental testbed. This will allow us to evaluate and improve its performance in large-scale scenarios with real users (i.e. with realistic mobility, social and traffic generation patterns). Moreover, the weights used by the social-based metrics criteria need to be automatically adjusted in real time, according to the network changes and conditions. This association model represents a first step in combining social-based metrics for association into mobile network scenarios.

C.7 Acknowledgment

This work was supported by the Fundação para Ciência e Tecnologia - FCT, through the grant SFRH/BD/62511/2009. The authors also thank the support provided by UBIQUIMESH project PTDC/EEA-TEL/105472/2008.

References

[AKET09] G. Athanasiou, T. Korakis, O. Ercetin, and L. Tassiulas. A cross-layer framework for association control in wireless mesh networks. *Mobile Computing, IEEE Transactions on*, 8(1):65 –80, jan. 2009.

[DH07] E. M. Daly and M. Haahr. Social network analysis for routing in disconnected delay-tolerant manets. In *Proceedings of the 8th ACM international symposium on Mobile ad hoc networking and computing*, MobiHoc '07, pages 32–40, New York, NY, USA, 2007. ACM. http://doi.acm.org/10.1145/1288107.1288113.

[FV11] F. Fabbri and R. Verdone. A sociability-based routing scheme for delay-tolerant networks. *EURASIP J. Wirel. Commun. Netw.*, 2011:1:1–1:13, January 2011. http://dx.doi.org/10.1155/2011/251408.

[HCY11] P. Hui, J. Crowcroft, and E. Yoneki. Bubble rap: Social-based forwarding in delay-tolerant networks. *Mobile Computing, IEEE Transactions on*, 10(11):1576 –1589, nov. 2011.

[IEE07] IEEE std. 802.11-2007, part 11: wireless LAN medium access control (MAC) and physical layer (PHY) specifications, June 2007.

[LDS03] A. Lindgren, A. Doria, and O. Schelén. Probabilistic routing in intermittently connected networks. *SIGMOBILE Mob. Comput. Commun. Rev.*, 7:19–20, July 2003. http://doi.acm.org/10.1145/961268.961272.

[LMY+11] Y. Le, L. Ma, H. Yu, X. Cheng, Y. Cui, M. A. Al-Rodhaan, and A. Al-Dhelaan. Load balancing access point association schemes for ieee 802.11 wireless networks. In *Proceedings of the 6th international conference on Wireless algorithms, systems, and applications*, WASA'11, pages 271–279, Berlin, Heidelberg, 2011. Springer-Verlag. http://dl.acm.org/citation.cfm?id=2033150.2033175.

[Lug08] G. Lugano. Mobile social networking in theory and practice. *First Monday*, 13(11), November 2008.

[MH11] J. Magalhaes and M. Holanda. Eiko: A social mobile network for manet. In *Information Systems and Technologies (CISTI), 2011 6th Iberian Conference on*, pages 1 –5, june 2011.

[MMDA10] A. Mtibaa, M. May, C. Diot, and M. Ammar. Peoplerank: Social opportunistic forwarding. In *INFOCOM, 2010 Proceedings IEEE*, pages 1 –5, march 2010.

[NDXT11] N. Nguyen, T. Dinh, Y. Xuan, and M. Thai. Adaptive algorithms for detecting community structure in dynamic social networks. In *INFOCOM, 2011 Proceedings IEEE*, pages 2282 –2290, april 2011.

[NS-10] NS-3. Manual ns-3.9, 2010. Available at `http://www.nsnam.org/docs/release/3.9/manual.html`.

[PP09] G. Palla and Pollner. Social group dynamics in networks. In T. Gross and H. Sayama, editors, *Adaptive Networks*, volume 51 of *Understanding Complex Systems*, pages 11–38. Springer Berlin / Heidelberg, 2009.

[QMK10] B. Qureshi, G. Min, and D. Kouvatsos. A framework for building trust based communities in p2p mobile social networks. In *Computer and Information Technology (CIT), 2010 IEEE 10th International Conference on*, pages 567 –574, 29 2010-july 1 2010.

Annex D

Multimedia Streaming Dissemination through Cooperative Device-to-Device Behaviors in Mobile Networks

Lucas Guardalben, Tomé Gomes, Paulo Salvador and Susana Sargento

Submitted to IEEE JSAC Special Issue on Device-to-Device Communications in Cellular Networks

The format has been revised.

Multimedia Streaming Dissemination through Cooperative Device-to-Device Behaviors in Mobile Networks

Lucas Guardalben, Tomé Gomes, Paulo Salvador and Susana Sargento

Abstract

Nowadays mobile multimedia entertainment is almost pervasive into the majority of mobile devices, leading people to access, store and disseminate a diversity of multimedia contents anytime and anywhere. The main challenge comes from achieving high successful delivery rates with low end-to-end delays for dissemination of multimedia streaming at last-hop devices. This article presents a novel approach to disseminate multimedia streaming merging technical with social behaviors and preferences to better perform device affiliations. Those technical and social preferences allow an efficient inference of Device-to-Device behaviors, aiming to create the possibility to maintain the best associations between devices through their behaviors and interactions. The potential to consider a real Device-to-Device behavior is presented as a solution to enhance the dissemination of multimedia streaming process over mobile infrastructureless networks. Through two proposed case studies: (1) Mobile Tour Guide and (2) Soccer Stadium with its own peculiarities and characteristics, we show how our proposed solution enhances the dissemination and reception of multimedia services beyond the baselines analyzed.

D.1 Introduction

The era of multimedia entertainment through applications and services, such as games, instant photos, movies, TV series, music shows, interactive video streaming and conferencing at anytime and anywhere, has brought a new interaction's experience to the users. Mobile devices became an interesting option to extend this type of services, which are currently most common in people's home, under stable network access infrastructures. To extend these services with a degree of quality to mobile devices is a very complex challenge.

Considering the current trend of mobile devices being part of the network infrastructure, it requires new mechanisms to allow digital streaming dissemination in mobile devices, offloading those demands from Telecom operators. The mobile applications that make it possible have to overcome challenges of system-level services, and also enhance the quality of service (QoS) and quality of experience (QoE) of the user. This can be achieved, for example, improving network lower levels (e.g. MAC layer) using social inspired techniques [Gea12a, Gea12b].

To enable dissemination of multimedia streaming over mobile networks, the communication between the devices is performed hop-by-hop until reaching the destination devices. Generally, this process is done by mobile routing protocols e.g. Optimized Link State Routing Protocol (OLSR) and Ad hoc On-Demand Distance Vector (AODV), gathering the proper information to fulfill their next-hop routing tables. The information gathered is exchanged without distinction in the network layer (e.g. network control information and data multimedia streaming flows), suffering from common problems, such as high end-to-end delays and overhead to maintain the routing tables. In case of dissemination of multimedia streaming, the challenge increases due to the size and volume of data content and the partitioning problem, causing the network to be split into several smaller isolated end-to-end groups or subnets. Moreover, the mechanisms for dissemination need to automatically ensure that all multimedia content will be disseminated for the largest number of devices as possible (e.g until the last-hop devices), with low end-to-end-delay and maximizing success delivery rates.

To overcome these constraints, solutions working at MAC layer are desirable because they enable to separate network control (e.g. maintenance of links and topology) at MAC layer, concentrating upper-layer resources to efficiently disseminate multimedia content. Moreover, new mechanisms and techniques need to be developed or re-adapted considering technical and social user behaviors in order to allow effective dissemination of multimedia streaming, following stringent levels of quality of experience to the users.

Based on those challenges, in this article we propose a novel approach to disseminate multimedia streaming, where technical and social behaviors are jointly identified, classified and used for infrastructureless network association. The network behavior is inferred by the devices at the MAC layer, such as the number of neighbor devices, available resources, connections quality and stability, and the number of devices interconnected. These parameters are weighted by technical and social criteria such as community membership and user's social network accounts (e.g. Facebook, Google+, Myspace, Twitter, etc). We use Genetic Algorithm (GA) to optimize the input combination of the technical weights leading to the best solution. Merging technical and social criteria allows for an efficient inference of mobile network behavior to be used in a multi-level aware service optimization.

The work previously developed in [Gea12a, Gea12b] considers only technical metrics at MAC Layer to be used in nodes association. In this article we propose to merge technical behavior with real social network information through the social relationship criterion. This will enhance the dissemination of multimedia streaming process in mobile networks with human preferences. Additionally, the partitioning problem has been addressed, enabling the devices to create their own communities, keeping their information automatically synchronized. Our approach is evaluated in two case studies: (1) Mobile Tour Guide and (2) Soccer Stadium, where it is compared against Wireless SPanning Tree (WSPT) [GGSS13] and Standard IEEE 802.11 [80297] protocols. The obtained results show that our solution enhances the overall reception quality experienced by the interested users, as well as the average load efficiency ratio, maintaining a low end-to-end delay.

The remainder of this article is organized as follows. Section D.2 presents a brief overview about the related work. Section D.3 presents examples of Device-to-Device applications, behaviors, interactions and scenarios. Section D.4 describes our proposed solution to disseminate multimedia streaming over infrastructureless scenarios. The results of our solution over two proposed case studies (1) Mobile Tour Guide and (2) Soccer Stadium, are depicted and discussed in Section D.5. Finally, Section D.6 concludes the article.

D.2 Related Work

In the literature there is a wide collection of mechanisms and techniques for dissemination of multimedia content over different architectures (e.g. centralized, hierarchical or fully distributed) and over heterogeneous access technologies. Dissemination of multimedia streaming over mobile networks has encountered many challenges, which are addressed by numerous mechanisms and techniques [LKPG11]. Most relevant techniques are related to coding [VWS11], overlay [KSHC12], end-to-end transport [GSC+96], routing [GK12] and cross-layer optimizations [MGKK12]. Most aforementioned approaches involve upper layers (e.g. Network to the Application layer), and fewer efforts enable lower layer adaptations (e.g. Physical or MAC layer). According to [LKPG11], approximately 65% of the techniques to realize video streaming dissemination over mobile networks in the literature are related to cross-layer optimizations. Network convergence scenarios, as envisioned in Next-Generation networks, induce complex topologies; efficient mechanisms to maintain the consistency of the topology will be a fundamental piece to ensure user stability and effectiveness over fully-distributed mobile topologies.

Another factor that influences the efficiency of dissemination of multimedia content over infrastructureless networks is the network partitioning problem. The probability of partitioning is particularly high in networks with user movements instability, low node density due to its transmission range, or through physical obstacles which prevent the nodes to forward information [LKPG11]. With the goal of keeping the multimedia flows when the network partitioning is detected, some solutions use delay-tolerant mechanisms [LN04], which are not effective when the subject is live streaming content. Nowadays, the trend is to consider human-centric behaviors to effectively disseminate multimedia streaming, enhancing the quality of service and experience of the users over mobile scenarios. Most of the solutions inspired by social and human behaviors consider only technical characteristics which are found in the field of routing for Delay Tolerant Networks (DTNs) [TLLL10, DXT09, MMDA10, HCY11, DH07], where the main idea is to use measurements based on devices experience to predict the movement and contacts repetition of nodes. The problem of choosing the best forwarders in DTNs is shown in [FV11], where the authors introduce sociability in routing, a novel routing scheme that selects a subset of optimal forwarders among all the nodes and relies on them for an efficient delivery. However, the metrics proposed in this work take into account information at network layer, capturing only frequency and type of nodes encounters. The work proposed in [LDS03] makes use of the observations of the non-randomness of mobility and, to improve routing performance, the authors propose probabilistic routing, PROPHET, which uses history of encounters and transitivity. However, it is not guaranteed that a node with a higher criterion will be encountered within reasonable time. In [HCY11] the authors seek to understand the human mobility in terms of social structures. The authors also emphasize that it is possible to detect characteristic properties of social grouping in a decentralized manner from a diverse set of real world traces. However, it is supposed to work with a hierarchical community structure which tends to adapt slowly to changing needs.

As observed, very few works consider human behaviors to enhance the dissemination of multimedia streaming process. In this paper we address the network partitioning problems, considering human behavior through technical information from devices as well as their relationships at social networks, enhancing the dissemination of multimedia streaming over infrastructureless networks.

D.3 Device-to-Device: Behaviors, Interactions and Scenarios

The main group of multimedia content consumers are humans themselves. Humans have different needs, contexts and requirements; therefore, devices that disseminate multimedia content need to consider the exact human preferences and behaviors. Infrastructureless cooperative networks are able to provide services to the surrounding neighbors by forming a common community. This community is composed by a group of persons using mobile devices, cooperating, sharing or disseminating common multimedia contents (e.g. voice and video-conferencing or watching a shared movie on their devices).

It is very important to acquire, process and consider the behavior and preferences of the users (e.g. identify the users relative location and movements patterns and/or instabilities, identify the interaction of the user and his/her device, preferred multimedia content, etc...). Moreover, identifying these behaviors and using that information to optimize the network routing will result in the improvement of the underlying multimedia delivery service. With the human behavior information, networks are able to adapt and re-adapt in order to maximize the quality of service and experience effectiveness of the users. Thus, networks based on human-centric concepts together with the underlying technical characteristics enable cooperative mobile networks, where it is possible to efficiently disseminate multimedia streaming anywhere and anytime.

In this article, we consider two study case scenarios: (1) Mobile Tour Guide, and (2) Soccer Stadium, in order to exemplify our proposed solution of multimedia streaming dissemination. The case study scenarios are described with more details in the following subsections.

D.3.1 Case Study (1): Mobile Tour Guide Scenario

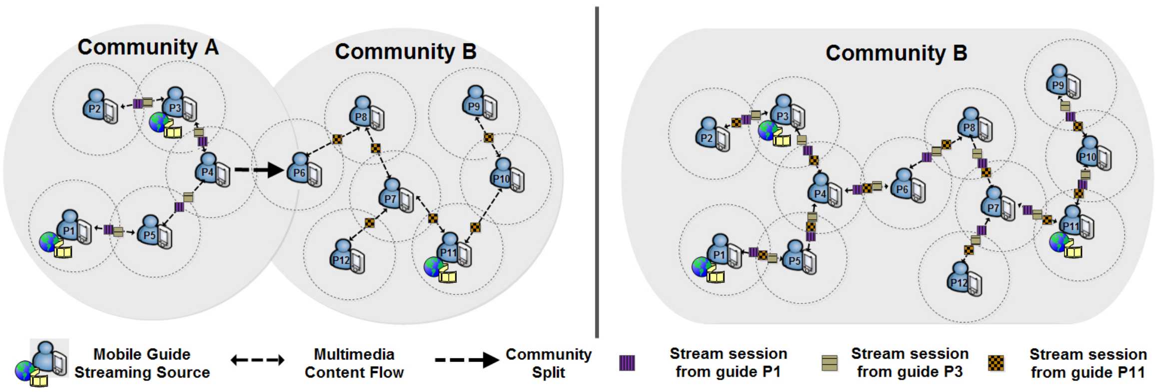

Figure D.1: Guided Tour: Union in communities of interest

Figure D.1 shows a guided tour use case. The visitors have the opportunity to access, watch and visualize multimedia streaming sessions from heterogeneous sources (e.g. Guides P1 and P3 from community A or Guide P11 from community B). The role of the communities in this scenario is to enable the visitors or guides to share common multimedia information. For example, with regard to the visitor P4, he has the possibility to join two different communities simply by being within the range of another visitor (e.g. P6), which belongs to the other community. Then, all visitors from both communities A and B through the visitor (P4) and (P6) will have the opportunity to receive streaming multimedia from P1, P3 and P11 which, now, are within the same community (B expanded).

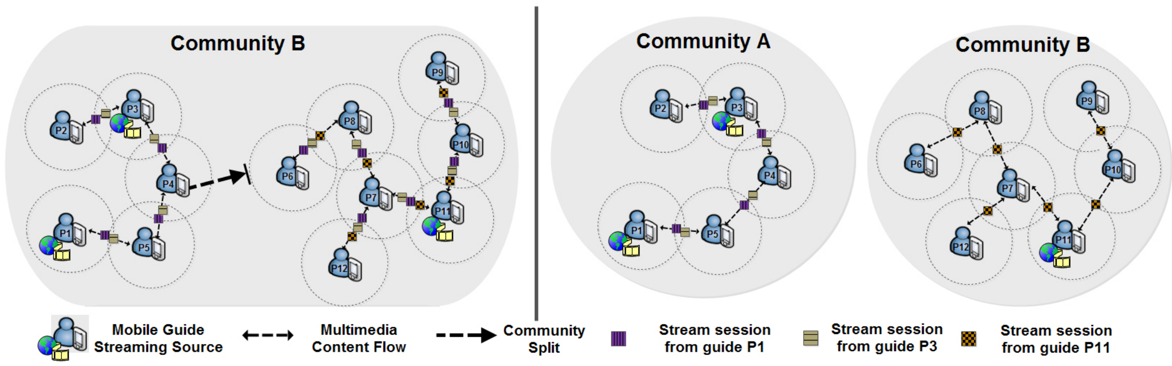

Figure D.2: Guided Tour: Split in communities of interest

Additionally, the visitors have the opportunity to choose the preferable video streaming to watch according to the progress of the tour (Figure D.2). For example, a guide aims to visit West part of the city (e.g Guide P1), and Guide P11 decides to go on East part. Then, the visitors have the possibility to follow the preferable Guide (P1 or P11) joining up into an interest community (e.g. A or B). Thus, a larger community can be separated into two or more different communities according to the visitors interests or preferences.

Inherent Challenges: Given the instability of the visitors and guide movements, the constant disconnections may appear, with impact in the efficiency of the dissemination of multimedia streaming. The dissemination process should be done quickly, maximizing the percentage of frames delivery (e.g. quality of the video) with low end-to-end delay, and taking into account the communities partitioning and multiple streams. Finally, the criteria to choose the best neighbors or communities to join must take into account the visitors mobile devices characteristics, behaviors, interactions, preferences and social relationships to increase the quality of service and experience for their visitors.

D.3.2 Case Study (2): Soccer Stadium Scenario

Figure D.3 shows an example of a soccer stadium where the soccer fans are organized in communities of interest (e.g. A and B). In the community A the soccer fans have the opportunity to visualize a specific multimedia streaming transmission (e.g. to watch and support the red team), while the community B can transmit specific streams of his opponent (e.g. blue team). Moreover, the live streaming source devices (e.g. P4 from community A, and P6 from community B) can be any soccer fans who are currently filming the game through their mobile devices with streaming video-recording capabilities, or can also be a soccer stadium infrastructure device that provides video streaming as an inherent service with professional narratives in real-time for their visitors. The soccer fan (P4) provides the multimedia streaming source to be transmitted to P3, P5, P2 and P1. Supporter P6 provides the different multimedia source stream session to soccer fans P8, P7, P11, P12, P10 and P9. In this scenario, the soccer fans movements are low; the communities are created according to the surrounding soccer fans, and its maintenance is minimal, practically static, except for special cases which occur mainly during the half-time.

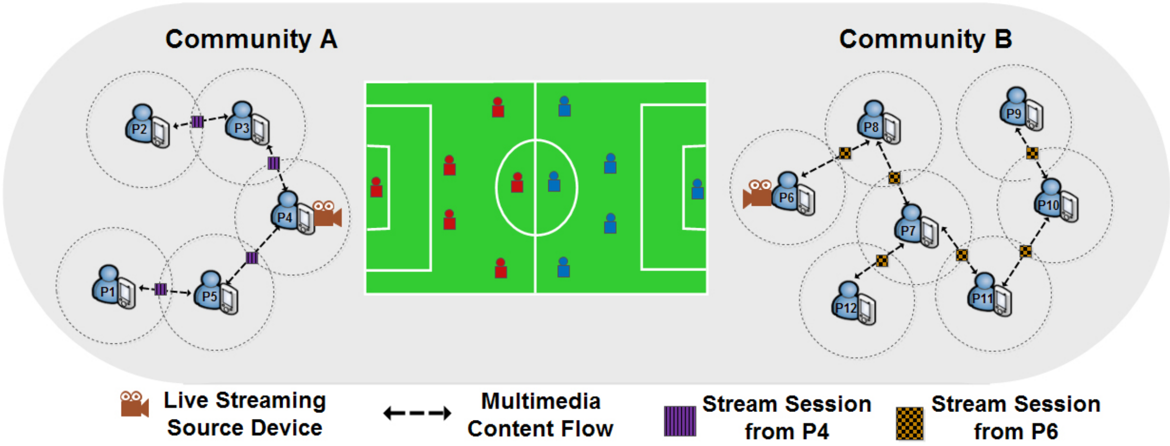

Figure D.3: Example of Soccer Stadium Scenario

Inherent Challenges: The biggest challenge here is to provide a well-defined criteria that considers the fans preferences and interactions in order to build-up a cooperative network. Thus, the challenge is to choose the best mobile device of a soccer fan from the whole surrounding ones which respects the requirements to forward efficiently the multimedia streaming sessions from the source to all last-hop destinations associated. Moreover, internal events of devices and resource utilization can disrupt the cooperation of a mobile device (e.g. a soccer fan receiving a call). In this case, the other soccer fans who use this device as forwarder should automatically find better alternatives to continue receiving the multimedia streaming.

D.4 Proposed Solution

Our proposed solution aims to create the possibility to maintain the best associations between devices through their behaviors and interactions. Considering the IEEE 802.11 MAC layer [IEE07] as the base standard and protocol used in this work, our proposed solution assumes a distributed mobile network consisting of devices that interact with each other through direct or indirect contacts, where devices have at least one common channel between their neighbors. Two main steps make part of our solution: (1) Devices Warm-up and Awareness, (2) Calculation of Device-to-Device Criteria. From a single device viewpoint, the warm-up process will set up the initial local information, e.g. internal identifiers, assignment of unique identifier and MAC address, local hardware status capabilities and setting up the communication interfaces. After this stage, the device is ready to start cooperating in the awareness process in order to identify the surrounding neighbor devices. We consider surrounding devices the ones that can communicate with each other in the transmission range. The second step comprises the classification and identification of technical and social behavior, proposing a set of well-defined criteria for companionship, partnership, membership and relationship, which will be used to choose the best device associations (see subsections D.4.1, D.4.2, D.4.3 and D.4.4).

As an initial step, a device must be aware of the surrounding context, identifying other devices available to relay information. Note also that the warm-up and awareness are complementary processes, working together to configure the device itself, as well as to start collecting the initial information of surrounding devices. The device awareness process builds and periodically updates the gathered information (e.g. from neighbors device), i.e. it searches for new devices and updates the information of neighbors. All the gathered information is stored in local tables at the device, denoted as *Partial View*, where each entry is filled with the neighbor information; it can also be updated according to topology changes, including neighbor device ID, timestamps, statistics from devices communication, number of exchanged messages and partnership criterion values (more details in the subsection D.4.2). Therefore, the *Partial View* is dynamic and its size varies according to the new received information. In addition, if the information is outdated (i.e. the device did not receive any contact in a predefined amount of time), it assumes that the device is no longer in range. Let's suppose that, in any given time instant, all devices have a *Partial View* of device i which is given by:

$$\mathscr{P}_i = \{j : j \in \mathscr{N} \wedge j \neq i \wedge v(i,j) = 1\}, \tag{D.1}$$

where $\mathscr{N}$ represents a set of mobile devices such that $\{i : i = 1, \ldots, N\}$, and $v(i,j)$ is a function that returns 1 if devices i and j are neighbors and 0 otherwise.

Moreover, the sub-set of nodes associated with node i can be defined by:

$$\mathscr{A}_i = \{j : j \in \mathscr{P} \wedge j \neq i \wedge a(i,j) = 1\} \tag{D.2}$$

where $a(i,j)$ is a function that returns 1 if nodes i and j are associated and 0 otherwise.

A community is a set of devices associated with each other, i.e, it is ensured that there is always an available path between any two or more devices. This concept is important from the point of view of forwarding multimedia streaming content in the community. The subset of nodes in the same community of node i can be defined by:

$$\mathscr{C}_i = \{j : j \in \mathscr{N} \wedge c(j) = c(i)\} \tag{D.3}$$

where $c(i)$ is a function that returns the identifier of the community of node i. Note that, $\forall i, \mathscr{C}_i \subseteq \mathscr{N}$ and $\forall i, |\mathscr{P}_i| \leq |\mathscr{C}_i| \leq |\mathscr{N}|$, where $|.|$ represents the cardinality of a sub-set.

On the other hand, the maintenance of the communities can sometimes be subject to partitioning, causing the community to be split into smaller isolated ones, limiting the end-to-end connectivity. Thus, the community partitioning problem can be defined as in the following:

Let i be a device that periodically sends messages to its surrounding neighbors ($\mathscr{P}_i$). In any given time instant this device can be connected (directly or indirectly) to the set of devices that belong to community $\mathscr{C}_i$. However, due to the people movement in different directions, weak device transmissions and receptions can be experienced, caused by, for example, physical barriers in the environment. This can lead to the partitioning of the network into several small groups. The constant communities maintenance, as well as the consistency of the information exchanged, need to be considered and to be seamless to the users.

In order to overcome the partitioning problem, in our solution, the communities are created and optimized in a distributed way, since each device is associated to the one it considers to be the best one in the surrounding neighborhood. In parallel, the process of maintaining the community takes place, since associations are not ensured to be definitive. The association is performed to the node with the highest Device-to-Device behavior factor $\mathcal{DTD}$ (see

Algorithm 1: Community membership information

```
/* node(i) should have at least one association                                  */
while node(i).recv_msg() do
    if (node(i) is Parent node) then
        /* Extracts membership reference value received from the child nodes     */
        /* Parent node(i) announces back to all child ∈ associated nodes          */
    else
        /* node(i) is a Child node                                                */
        /* Child node(i) reports its local average membership value to the parent node */
    end
end
```

Subsection D.4.4) in the vicinity. Each time a device is contacted by another one whose factor is higher than the one of its current association, it needs to signal the device which holds the current connection, indicating that it will end the association. The loss of connection can occur not only at the extremes of the community, but also at the core of the community. One community will maintain its identifier, and the newly created one will have to adopt a new ID. Therefore, the device that loses the connection will be responsible for generating a new identifier and spread it to the devices associated, until all devices agree on the current community identifier. In order to maintain the communities information, a distributed algorithm (1) is proposed where: (i) all devices report to their parent node the information about the nodes associated to them; (ii) the parent node (i.e. the first node of the community and overall parent) processes all reports from its child nodes, estimates the overall size and quality of the community, and announces them back to all child nodes; and (iii) all nodes report to their child nodes the information announced by the parent node. Moreover, the need to propagate information between nodes on a distributed community justifies the importance of the proposed algorithm. Additionally, the size and quality of the community serve as input to calculate the Community Membership criteria (see Subsection D.4.3).

The next subsections will define the proposed Device-to-Device behavior through technical and social information in order to provide the best nodes' affiliations.

D.4.1 Crowd Companionship Criterion

The crowd companionship criterion: (1) enables a person to be associated with another person which has the largest density of surrounding people in its coverage area; and (2) defines degrees of personal communication. So, combining both these factors allow the devices to decide the best companionship (association with a neighbor device) to establish an association. In Figure D.4 for example, person P1 is within range of persons P2 and P3. Then, the quality levels represented by the *Qlevels* table ranging from *Good* (1 - green color) to *Bad* (4 - red color) will determine the levels of personal communication. Following this example, P3 has better ranking for person P1, so a companionship between them will be performed.

In practice and analytical terms, the density of devices surrounding neighbor $\boldsymbol{j}$ relatively to device $\boldsymbol{i}$ is given by $\boldsymbol{\mathcal{D}_{i,j}} = \left(\frac{|\mathscr{P}_j|+1}{|\mathscr{P}_i|+1}\right)$, where $|\boldsymbol{\mathcal{P}_i}|$ represents the cardinality of the set of neighbors of device $\boldsymbol{i}$, i.e., the size of the neighborhood of device $\boldsymbol{i}$. The personal communication is given by $\boldsymbol{\mathcal{U}_{i,j}} = \left(\frac{h_{i,j}}{\sum_{k:k\in\mathscr{P}_i} h_{i,k}}\right)$, where $\boldsymbol{h_{i,j}}$ represents the number of Hello messages received by device $\boldsymbol{i}$ from device $\boldsymbol{j}$ in the last $\boldsymbol{\Delta}$ seconds. The denominator $\sum_{k:k\in\mathscr{P}_i} h_{i,k}$,represents the sum of all Hello packets received in the last $\boldsymbol{\Delta}$ seconds.

The overall crowd criterion of a device $\boldsymbol{i}$ can then be inferred averaging the personal communication between device $\boldsymbol{i}$ and all other devices in its neighborhood ($\mathscr{P}_i$):

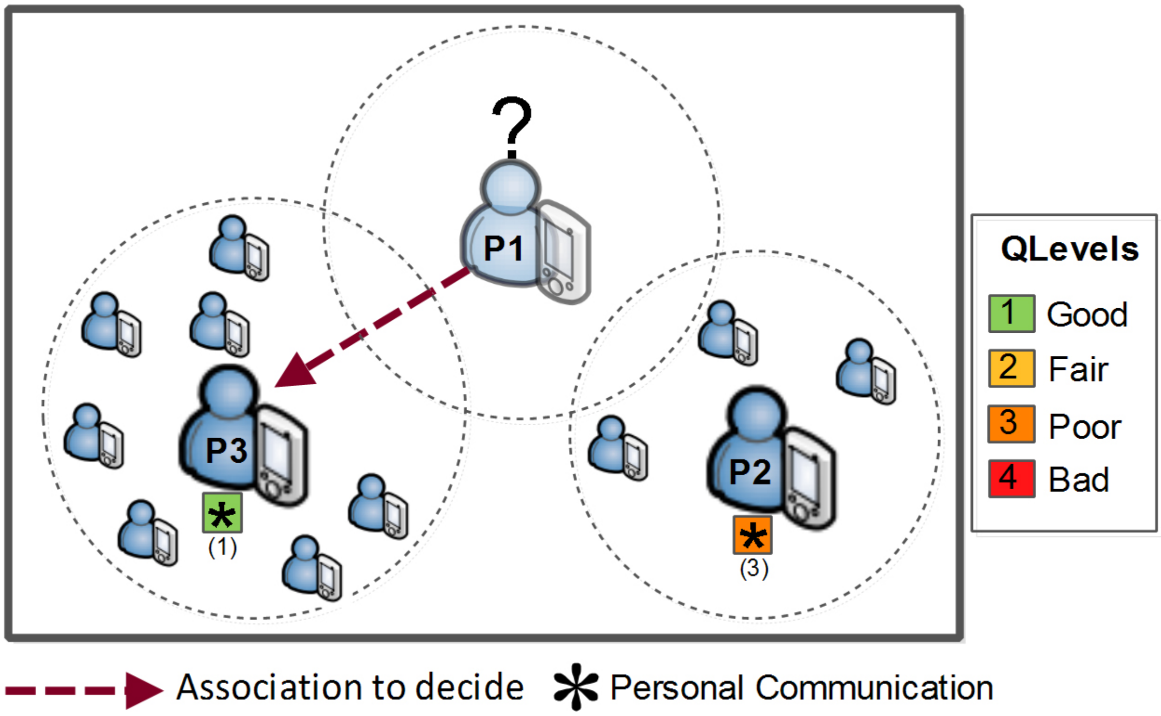

Figure D.4: Crowd behavior example

$$\overline{\mathcal{CC}_i} = \frac{1}{|\mathscr{P}_i|} \sum_{j:j \in \mathscr{P}_i} (\mathcal{D}_{i,j} \times \mathcal{U}_{i,j}) \quad \text{(D.4)}$$

Note that, after a companionship to be formed (e.g P1 to P3), P1 is able to contribute with its resources in order to cooperate to create other companionships.

D.4.2 Person-to-Person Partnership Criterion

The person-to-person partnership criterion resembles the idea of the crowd companionship criterion, but the factors involved to decide the partnerships are different: person device capabilities and utilization, person environment and person stability. The non-intrusive possibility to measure the capabilities and utilization of a person device is interesting because it can infer if a person mobile device has all the necessary requirements. Through person environment it is possible to get information for example, if there are many physical obstacles that constrains the person movements. Finally, the person stability estimates if the person moves closer/farther over dense/sparse areas.

In Figure D.5, person P1 has two possibilities to create a partnership in the coverage area (e.g with P2 or P3). The person P2 has poor device features, fair in terms of interference, maybe caused by physical obstacles and bad stability. Since P3 has better levels within the features analyzed, P1 will choose the person P3 to create a partnership. In practical terms, the user device capabilities and usage takes into account local device resources (e.g battery remaining, CPU/Memory status and storage capacity). The person environment measures the impact of indoor or outdoor physical obstacles interferences, and the person stability taking into account the mobility of the person. Following the example, P2 will not be a good candidate to create partnerships.

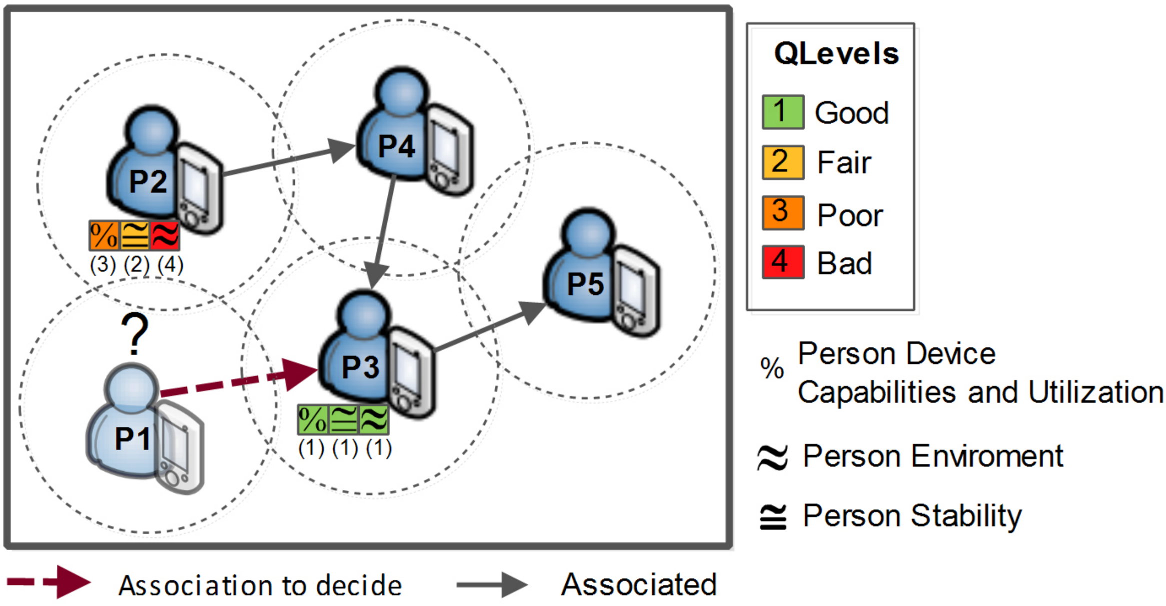

Figure D.5: Person-to-person behavior example

In analytical terms, the person-to-person criterion between device $\boldsymbol{i}$ and $\boldsymbol{j}$ with an active association can then be inferred averaging the quality of device $\boldsymbol{i}$ and all devices associated ($\mathscr{A}_i$), and is given by:

$$\overline{\mathcal{PP}_i} = \left(\frac{1}{|\mathscr{A}_i|} \sum_{j:j\in\mathscr{A}_i} (\mathcal{R}_j \times \rho_{i,j} \times \mathcal{S}_{i,j}) \right) \tag{D.5}$$

where $\mathcal{R}_j$ quantifies device $\boldsymbol{j}$ available resources characteristics (such as processor, memory, storage, battery remaining and type of interfaces). $\rho_{i,j}$ is an estimate of the user environment through the Signal to Noise Ratio (SNR) between devices $\boldsymbol{i}$ and $\boldsymbol{j}$. The user stability $\mathcal{S}_{i,j}$ is estimated by the ratio between the number of Hello messages received by device $\boldsymbol{i}$ from a node $\boldsymbol{j}$, and the total number of Hello messages sent by device $\boldsymbol{j}$ (s_j):

$$\mathcal{S}_{i,j} = \frac{h_{i,j}}{s_j}. \tag{D.6}$$

The value from $\overline{\mathcal{PP}_i}$ is transmitted through Hello contact messages between devices.

D.4.3 Community Membership Criterion

People are able to join or leave a community anytime. It is important to ensure a way to gather the overall community information in order to provide different options for a person to become a member of the desired community of interest. In both aforementioned criteria, the information analyzed to decide the best partnership is gathered from its surrounding nodes. It is important to report the status of the communities in terms of quality of devices and their communications to the person who wants to create a membership. Following the example in the Figure D.6, person P6 has in its coverage area persons P5 and P8. In any given moment, the information about the quality of the entire community is transmitted hop-by-hop between the person through his mobile devices, and the future member (e.g. P6) will be aware of the current quality of communities.

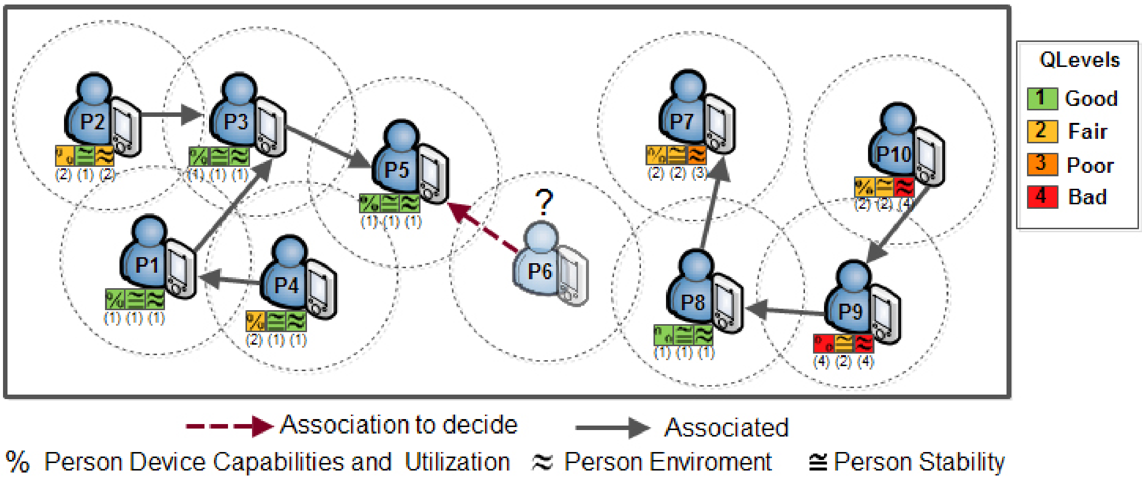

Figure D.6: Community behavior example

For example, since P1, P2, P3, P4 and P5 belong to the community A, and P7, P8, P9, P10 belong to community B, it is not clear how P6 has the possibility to choose a community to participate. To become a member of a community, P6 needs to establish communication with the surrounding neighbors to evaluate the communities available. In our example, P5 will be the node responsible to send the overall information to P6 about the community A, and P8 about the community B. In this case, both nodes P5 and P8 satisfy the requirements of P6. However, the overall information disseminated about the communities quality induces P6 to decide to associate to community A. In analytical terms, the idea is to give more value to communities that have more nodes and higher person-to-person criteria values. The community membership criterion quantifies the size of the community and the quality of the relations within it, and is given by:

$$\mathcal{CM}_i = \sum_{j:j\in\mathscr{C}_i} \overline{\mathcal{PP}_j} \tag{D.7}$$

where $|\mathscr{C}_i|$ represents the cardinality of the set of nodes in the same community as node $\boldsymbol{i}$, i.e., the number of nodes in the same community of node $\boldsymbol{i}$.

At this point, we have identified and classified the technical behaviors, and the final value is obtained by performing a weighted sum of the three criteria: companionship, partnerships and membership, which is given by:

$$\mathcal{CPM}_{i,j} = \left(w_1\overline{\mathcal{CC}_i} + w_2\overline{\mathcal{PP}_i} + w_3\mathcal{CM}_i\right) \tag{D.8}$$

where $\boldsymbol{w_1}, \boldsymbol{w_2}$ and $\boldsymbol{w_3}$ are the respective weights of each criteria.

D.4.4 Social Relationship Criterion

Social networks have revolutionized interpersonal social relationships, allowing to meet or contact friends everyday. The ability to interact with friends listed in these social networks is an interesting option for people to share their hobbies, musical tastes, preferences or post comments or photos, among other functionalities.

The social relationship criterion takes into consideration the information obtained from social networks, identifying and quantifying the amount of human similarities. This metric

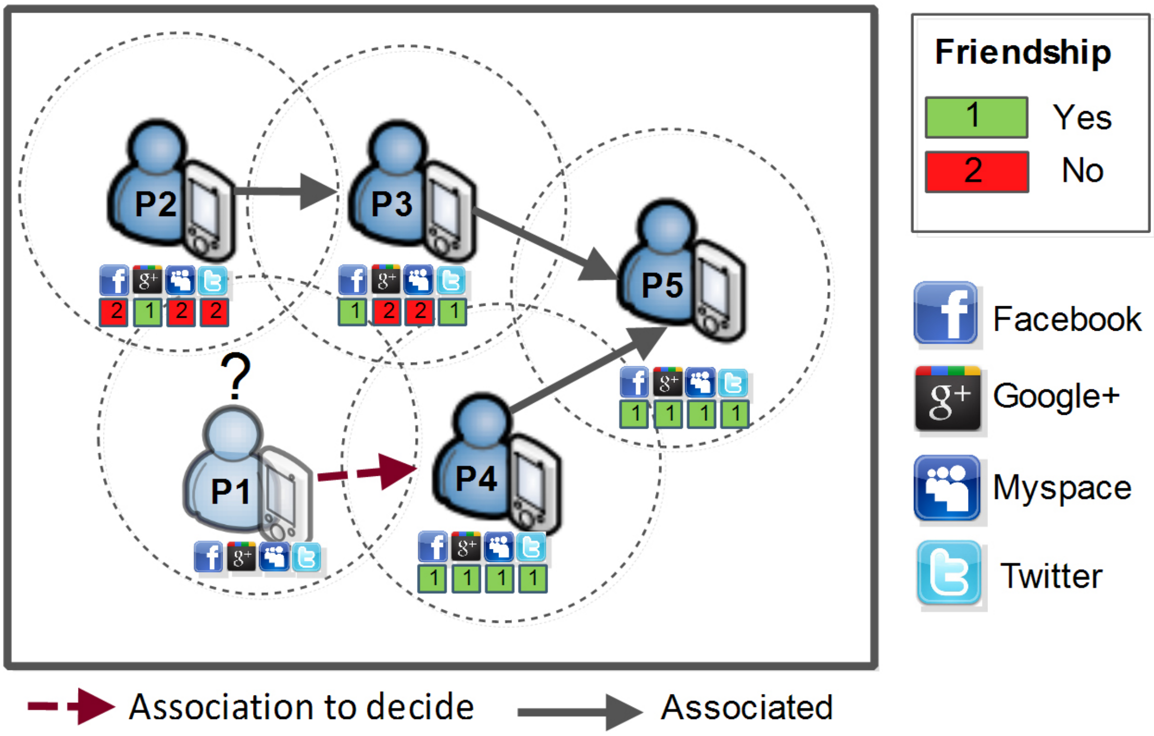

Figure D.7: Social network relationship example

is inferred by identifying the number of common social network relations between persons (e.g. Facebook, Google+, Myspace, Twitter, etc). Following the example in the Figure D.7, person P1 has in its coverage area person P2, P3 and P4, where P4 has more social network connections; thus P1 decides to associate to P4. In analytical terms the social relationship criterion is given by:

$$\mathcal{FF}_{i,j} = \left(\frac{\mathcal{SN}_j + 1}{\mathcal{SN}_T + 1}\right) \tag{D.9}$$

where $\mathcal{SN}_j$ represents the number of social networks the person on node j is friend with the person on node i, and $\mathcal{SN}_T$ represents the total number of social networks considered as reference.

Therefore, the final value of the proposed Device-to-Device behavior is given by:

$$\mathcal{DTD}_{i,j} = \mathcal{CPM}_j \times \mathcal{FF}_{i,j} \tag{D.10}$$

where the inferred technical behavior from node i can be merged with the human social relationships between the user on node i and the user on node j, enabling a complete Device-to-Device behavior quantifier between node i and j.

In the next section, we show the influence of Device-to-Device behavior, demonstrating the efficiency and effectiveness for the dissemination of multimedia streaming.

D.5 Simulation and Results

This section presents a simulation study that evaluates our proposed solution and its impact on the dissemination of multimedia content over two case studies: (1) Mobile Tour

Guide and (2) Soccer Stadium. The evaluations were carried out in Network Simulator v.3.9 (NS-3) [NS-10]. We have considered *Nomadic* and *Waypoint* as mobility patterns for the first case study, and *Static* and *Edge* for the second one. In *Nomadic* mobility pattern, the nodes move to random destination points within the vicinity of a mobile reference node, where the reference node moves randomly. This pattern mimics the mobility of a guided tour where people move towards a tourist guide. In *Waypoint* mobility pattern, all nodes move towards a point belonging to a sequence of random generation points. This pattern emulates an environment with a sequence of meeting points. The *Edge* mobility pattern rapidly moves the nodes to the border of the simulation area where they stop. With the *Static* pattern the nodes are randomly placed within the simulation area and never move.

The results presented in this section include the mean values of 10 independent repetitions with confidence intervals of 95%. The performed simulations are 60s long and the network size varies from 10 to 40 nodes. The simulation window is 150m x 150m and the coverage radio is 50 meters.

D.5.1 DtD:Optimized

In this article we use a Genetic Algorithm (GA) to optimize the experiments performed in NS-3 simulator. The problem to be solved is the optimization of weights $\boldsymbol{w_1}$, $\boldsymbol{w_2}$ and $\boldsymbol{w_3}$ of the technical critera $\overline{\mathcal{CC}_i}$, $\overline{\mathcal{PP}_i}$ and $\mathcal{CM}_i$, improving the decision of nodes' association.

Regarding to the optimization problem, we consider a discrete-event simulation model with n deterministic input parameters: $\boldsymbol{\mathcal{W}} = \{\boldsymbol{w_1}, \boldsymbol{w_2}, ..., \boldsymbol{w_n}\}$, and k stochastic output variables: $\boldsymbol{\varphi} = \{\boldsymbol{\varphi_1}, \boldsymbol{\varphi_2}, ..., \boldsymbol{\varphi_k}\}$, where $\boldsymbol{\varphi}$ is a function of $\boldsymbol{\mathcal{W}}$: $\boldsymbol{\varphi} = \boldsymbol{f}(\boldsymbol{\mathcal{W}})$. We assume that $\boldsymbol{\mathcal{W}}$ is a feasible region where the input parameters are defined, and that $\boldsymbol{\Lambda}(\boldsymbol{\varphi})$ is defined as a real function that aggregates the k output variables into a single one. The objective of the overall optimization process is to determine the values for $\boldsymbol{\mathcal{W}}$ such that $\boldsymbol{f}(\boldsymbol{\mathcal{W}})$ leads to the optimal simulation response, i.e., $\boldsymbol{f}(\boldsymbol{\mathcal{W}}) \equiv \boldsymbol{E}[\boldsymbol{\Lambda}(\boldsymbol{\varphi})]$, where $\boldsymbol{E}[\boldsymbol{\Lambda}(\boldsymbol{\varphi})]$ corresponds to the optimal output value. The challenge to determine the optimal $\boldsymbol{\mathcal{W}}$ set is that $\boldsymbol{f}(\boldsymbol{\mathcal{W}})$ cannot be observed directly, i.e, it cannot be calculated deterministically because it varies over time, so it must be estimated. This may require several replications and long simulation runs.

Our main target is to obtain an optimal solution for the input parameters related to technical behaviors that directly influence the chosen links between nodes and, therefore, the network topology of a complex time-variant system implemented in the discrete-events simulator NS-3. The goal is to optimize the network regarding two measurable outputs: maximize percentage of multimedia reception quality, and minimize the distance to the multimedia content generators. This means that we are aiming to improve the number of nodes that are able to receive the multimedia streaming, using multiple hops in the minimum amount of time, considering that the number of hops is directly proportional to the delay introduced in the communications.

The main reasons that distinguish GAs from other optimization methods are summarized in [Gol89]: the set of input parameters is encoded and not directly manipulated; they search from a set of points rather than a single one, allowing the improvement of its convergence; payoff (fitness) is used to evaluate each solution and not derivate functions or other auxiliar knowledge, turning it into an easy-to-use black-box; and the use of probabilistic instead of deterministic rules, which reduces the propability of being trapped in a local optimum. Figure D.8 illustrates the interaction between GA, the NS-3 and the fitness function. The GA interacts with NS-3 using multiprocessing, requesting simultaneous simulations with different input values (weights $\boldsymbol{w_1}$, $\boldsymbol{w_2}$ and $\boldsymbol{w_3}$). Next, the output of each experiment is analysed in order to evaluate the result in terms of dissemination percentage $\boldsymbol{p}$ and distance $\boldsymbol{d}$ (the outputs that are expected to be maximized and minimized, respectively). These will be the inputs to the fitness function, which is utilized to determine the corresponding fitness value at the end of each experiment (i.e. of each combination of weights). Thereupon, the resulting value is added to a pool that will lead to a new generation. The higher the experiment fitness, the higher the probability of the corresponding combination of weights to be selected to the next generation.

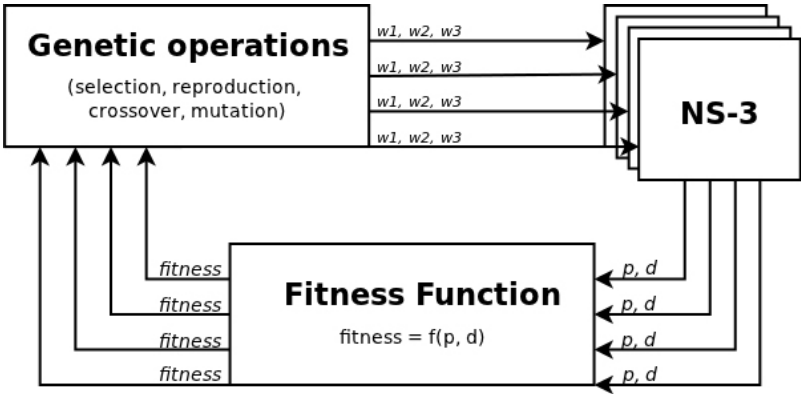

Figure D.8: Interaction between Genetic Algorithm and NS-3

However, the value of the selected inputs $\boldsymbol{w_1}$, $\boldsymbol{w_2}$ and $\boldsymbol{w_3}$ cannot be directly manipulated, but rather they must be encoded into chromosomes. Several schemes can be used (binary, permutation, tree, etc), being the binary the chosen one in this solution. Although this coding scheme cannot be directly applied to many problems, using enough bits for precision and a simple conversion from the natural system units (integer, float, etc) to binary should be sufficient to use the scheme. In the presented problem, knowing that the input weights $\boldsymbol{w_1}$, $\boldsymbol{w_2}$ and $\boldsymbol{w_3}$ can vary between 0 and 100%: $\sum \_ \boldsymbol{i = 1^3 w_i = 1}$, the chromosomes have been designed with 21 bits. Therefore, each weight takes 7 bits, so that there are enough bits to meet the precision requirements $\boldsymbol{(\frac{max\%}{2^{bits}} = \frac{100}{2^7} \approx 0.8\%)}$, thus it is possible to vary each weight in steps of 0.8%).

Taking into account the required number of bits for the chromosomes and other parameters related to the GA behavior, the optimization method was tuned to our solution (Table D.1).

Table D.1: Genetic algorithm parameters

Parameter	**Value**
Chromosome bits	21
Population size	20
Max generation	100
Mutation prob.	0.1
Crossover prob.	1

Although the genetic operators have an important role in the optimization evolution, the element which indicates the degree of optimization is the fitness function. Therefore, it is crucial that it is designed to describe the objective that it aims to model with very high accuracy. Specifically to this problem, it is known that the goal is to maximize the reception quality through the percentage of multimedia content disseminated to the nodes, and at the same time minimize the distance to the multimedia content generator nodes. Therefore, the fitness function should benefit higher values of reception quality percentage (positive exponent) and negatively affect the hops distance (negative exponent).

This leads to the normalized formula:

$$\boldsymbol{fitness(p,d)} = \begin{cases} \frac{\mathbf{100}*e^{p}+(-e^{d})}{\mathbf{100}*e^{1}+(-e^{0})} & \text{, if p} > \ln\left(\frac{e^{d}}{\mathbf{100}}\right) \\ \mathbf{0} & \text{, if } \boldsymbol{p} \leq \ln\left(\frac{e^{d}}{\mathbf{100}}\right) \end{cases} \quad \text{(D.11)}$$

where $\boldsymbol{p}$ is the percentage of multimedia reception quality [0:1] and $\boldsymbol{d}$ is the distance to the multimedia content generator node. The value of $\boldsymbol{p}$ in the condition is the one that prevents the fitness function to return negative values.

D.5.2 Simulation Results

In our simulations we consider only multimedia traffic between the nodes. At any instant, any node may start a multimedia streaming. The broadcasting nodes and starting time are randomly chosen. The multimedia streams have a rate of 10 packets/s and an exponential random duration. Moreover, the users may or may not express interest in the video streaming transmission. We assume that a user associated with a friend at social network that is generating the multimedia stream has a higher probability (80%) of interest in the content. The stream interest of a node is randomly generated upon the start of a new stream.

The evaluation results for $\boldsymbol{\mathcal{DTD}}$ will include two different methods for determining the weights for $\boldsymbol{\mathcal{CC}_i}$, $\boldsymbol{\mathcal{PP}_i}$ and $\boldsymbol{\mathcal{CM}_i}$: (i) optimized through GA ($\boldsymbol{\mathcal{DTD}}$:Optimized) and (ii) pre-determined with balanced influence of **1/3** for each weight ($\boldsymbol{\mathcal{DTD}}$:Balanced).

As a baseline for the results, we consider the standard IEEE 802.11 [80297] and Wireless Spanning Tree (WSPT) [GGSS13] protocols.

In IEEE 802.11 the nodes will associate with the neighbor that makes the first contact; in WSPT the association has inherent hierarchical node organization favoring the lowest cost path to a community root (node with the lowest ID).

In the results, we have considered three evaluation metrics: (i) stream reception quality, which quantifies the percentage of successful delivered streaming packets (i.e. higher is better); (ii) distance to stream source, which quantifies the average number of hops to the stream source node, and which can be considered proportional to the transmission delay added to the stream packets (i.e. lower is better); and (iii) average load efficiency ratio which quantifies the total number of multimedia packets divided by the average number of contacted nodes per multimedia flow (i.e. lower is better).

We have evaluated the MAC-Layer signalling cost which includes beacons, association and disassociation packets for all mobility patterns. As can be seen in Table D.2, the signalling cost of each protocol is expressed in megabytes (Standard, WSPT and $\boldsymbol{\mathcal{DTD}}$:Optimized) as a function of the number of nodes and mobility patterns. The extra information added to calculate the Device-To-Device behaviors on beacons and associations in the $\boldsymbol{\mathcal{DTD}}$:Optimized, makes the cost higher in most of the mobility patterns. This can be explained by the fact the $\boldsymbol{\mathcal{DTD}}$:Optimized always seeks the best criteria for association and frequently triggering request, response and disassociations packets to optimize the association criteria. In case of Standard and WSPT protocols, the signalling packets are triggered only when the devices are out of the coverage area, or if the Root ID changes. Finally, it is observed that the signalling cost per device in Mbits/sec of the $\boldsymbol{\mathcal{DTD}}$:Optimized is significantly small.

D.5.3 Mobile Tour Guide Scenario

Regarding the first criteria evaluated, i.e. the quality of the received multimedia stream, we observe that the $\boldsymbol{\mathcal{DTD}}$:Optimized and $\boldsymbol{\mathcal{DTD}}$:Balanced solutions outperform the other two approaches (Figure D.9). In the case of $\boldsymbol{\mathcal{DTD}}$:Optimized, it is possible to observe that

Table D.2: MAC Layer Signalling cost

Nomadic	Standard (MB)	WSPT (MB)	DtD:Optimized (MB)	(Mbits/s/dev)
10 nodes	0.63 ± 0.03	1.98 ± 0.16	0.66 ± 0.20	0.0088
20 nodes	7.88 ± 4.62	5.30 ± 0.48	16.02 ± 2.43	0.1068
30 nodes	17.48 ± 8.02	15.70 ± 1.58	32.17 ± 1.06	0.1429
40 nodes	28.46 ± 4.60	14.19 ± 1.01	35.92 ± 1.92	0.1197
Waypoint	Standard (MB)	WSPT (MB)	DtD:Optimized (MB)	(Mbits/s/dev)
10 nodes	0.37 ± 0.05	0.50 ± 0.10	0.60 ± 0.14	0.0080
20 nodes	1.54 ± 0.90	1.24 ± 0.23	11.87 ± 2.28	0.0791
30 nodes	1.03 ± 0.08	2.25 ± 0.56	19.14 ± 4.13	0.0850
40 nodes	1.28 ± 0.10	3.79 ± 0.60	11.00 ± 2.48	0.0366
Edge	Standard (MB)	WSPT (MB)	DtD:Optimized (MB)	(Mbits/s/dev)
10 nodes	0.50 ± 0.02	0.94 ± 0.05	0.14 ± 0.01	0.0010
20 nodes	1.36 ± 0.23	3.31 ± 0.18	0.50 ± 0.06	0.0033
30 nodes	2.49 ± 0.16	5.53 ± 0.22	1.80 ± 0.19	0.0080
40 nodes	6.16 ± 0.58	8.70 ± 0.45	8.70 ± 3.02	0.0289
Static	Standard (MB)	WSPT (MB)	DtD:Optimized (MB)	(Mbits/s/dev)
10 nodes	0.27 ± 0.02	0.38 ± 0.03	0.07 ± 0.02	0.0093
20 nodes	0.86 ± 0.10	1.06 ± 0.14	1.51 ± 0.49	0.0100
30 nodes	1.36 ± 0.11	2.28 ± 0.31	29.13 ± 8.73	0.1294
40 nodes	2.05 ± 0.29	3.02 ± 0.31	45.21 ± 0.77	0.1507

the gain of the reception quality is more than doubled compared to the baselines. Also, unlike the Standard solution, which is not able to adapt its first link choice unless the node goes out of range, the $\mathcal{DTD}$:Optimized solution scales well with the network size.

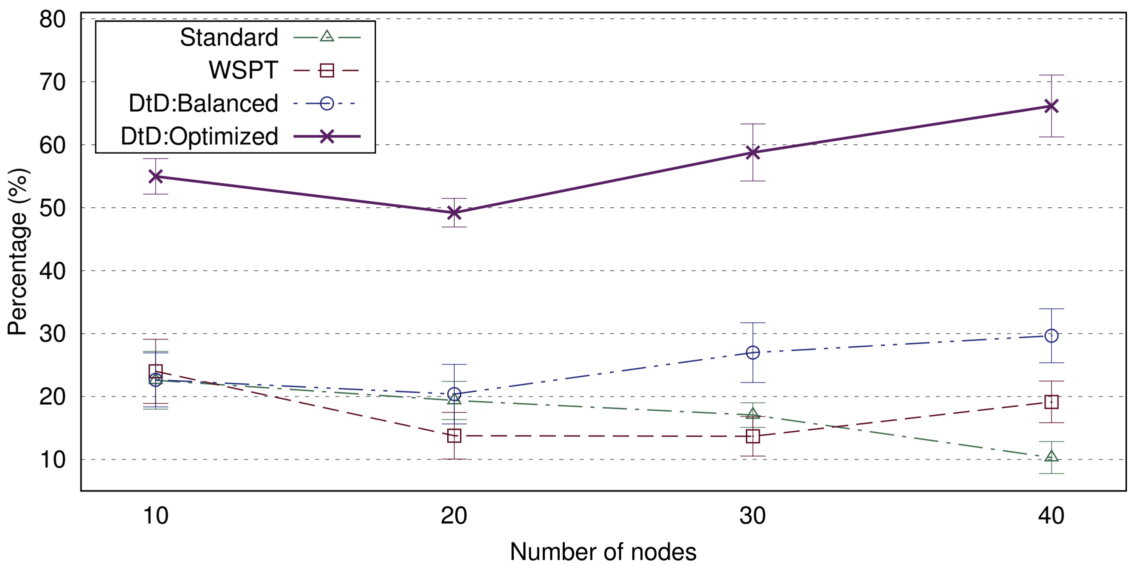

Figure D.9: Reception quality of stream in mobile tour guide environment for the *Nomadic* mobility pattern

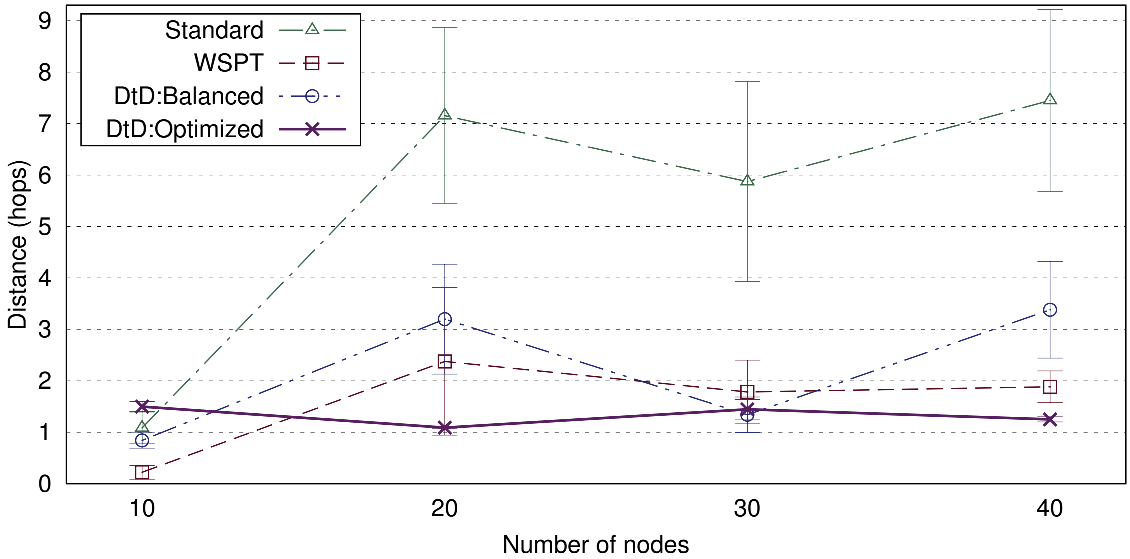

Figure D.10: Distance to stream source in mobile tour guide environment for the *Nomadic* mobility pattern

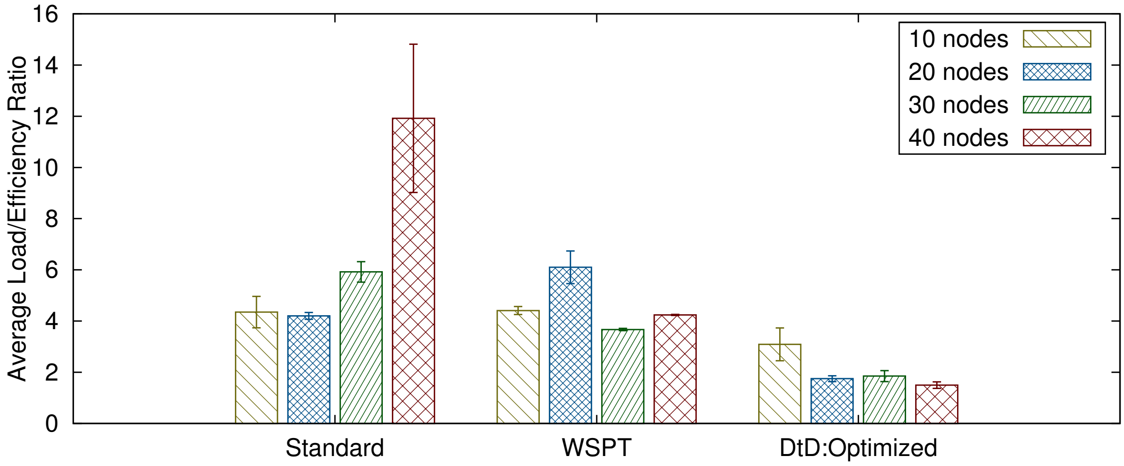

Figure D.11: Average load efficiency ratio in mobile tour guide for the *Nomadic* mobility pattern

The WSPT solution attempts to maintain the minimum cost path towards the elected root under constant changes of the network topology (in which even the root can be a subject to frequent changes), by temporarily disabling the communication between nodes. Taking into account the distance between the stream source and the nodes interested in receiving the multimedia stream, it is possible to assume that a higher number of relay nodes between those two will increase the packets delay.

Therefore, the results in Figure D.10 indicate second criteria evaluated i.e distance to stream source, that the $\mathcal{DTD}$:Optimized solution minimizes the path distance to the stream source node, similarly to the WSPT protocol, hence minimizing the delay between packets. It is important to note that, although the links between the nodes imply similar distance to the stream source in these two approaches, the $\mathcal{DTD}$:Optimized solution is able to deliver higher reception quality to users. This proves that it is not sufficient to base the criteria of establishing the link between the nodes solely on identifiers and number of hops to root (as in WSPT).

Regarding the third criteria evaluated, i.e. the average load efficiency ratio, the $\mathcal{DTD}$: Optimized shortens the distances between the multimedia source to destination, which improves the average load efficiency ratio, as can be seen in the Figure D.11. In spite of the Standard and WSPT solutions, the average load ratio became less efficient than the $\mathcal{DTD}$:Optimized, due to the higher distance (see Figure D.10) and lower dissemination reception (see Figure D.9), thus, more non-delivered packets remain into multimedia streaming flows.

The Figures D.12 and D.13 represent the Mobile Tour Guide environment through *Waypoint* mobility pattern. In the first figure, the results show that the $\mathcal{DTD}$:Optimized can improve the average quality received as opposed to the solution that uses equal weights, almost without affecting the distance to the source (Figure D.13). Although the reception quality between WSPT and the Standard approach are nearly the same, the WSPT is always able to establish the shortest possible path (Figure D.13).

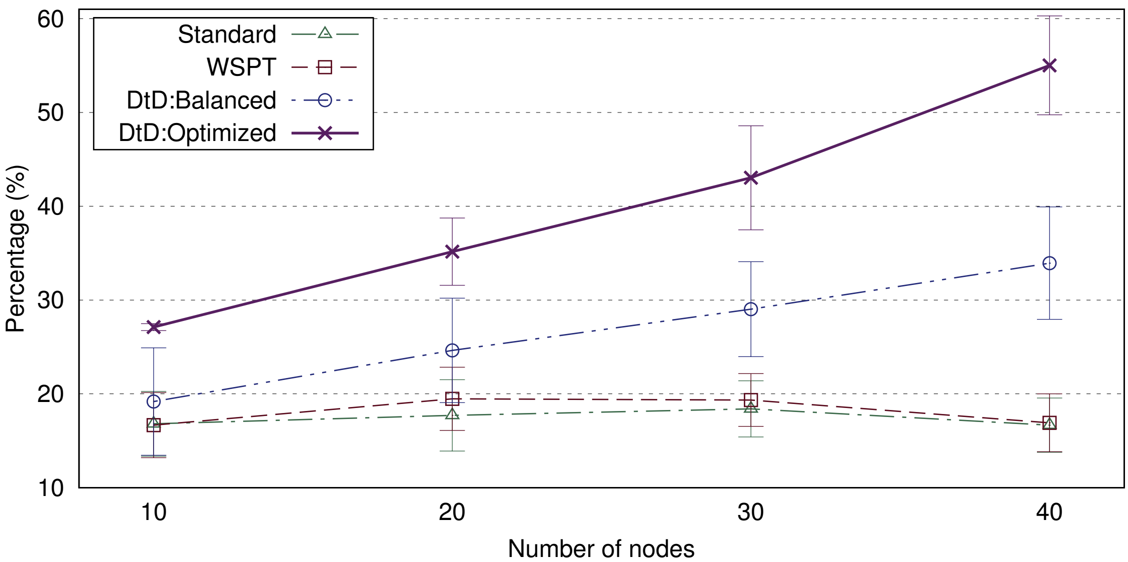

Figure D.12: Reception quality of stream in mobile tour guide for the *Waypoint* mobility pattern

The average load ratio evaluation follows the same behavior of the *Nomadic*, which reduces the amount of multimedia packets per average flow (see Figure D.14). Note that even with the larger distance from multimedia source generator (Figure D.13), the $\mathcal{DTD}$:Optimized compensates the average load ratio by the higher rate of packets delivery, as can be noted in Figure D.12.

D.5.4 Soccer Stadium Scenario

Scenarios without mobility were considered as well, i.e. using static nodes or nodes placed at the border of the environment area. Since nodes are randomly placed in the simulation area, the nodes interested in the multimedia stream can be too close or too far away from

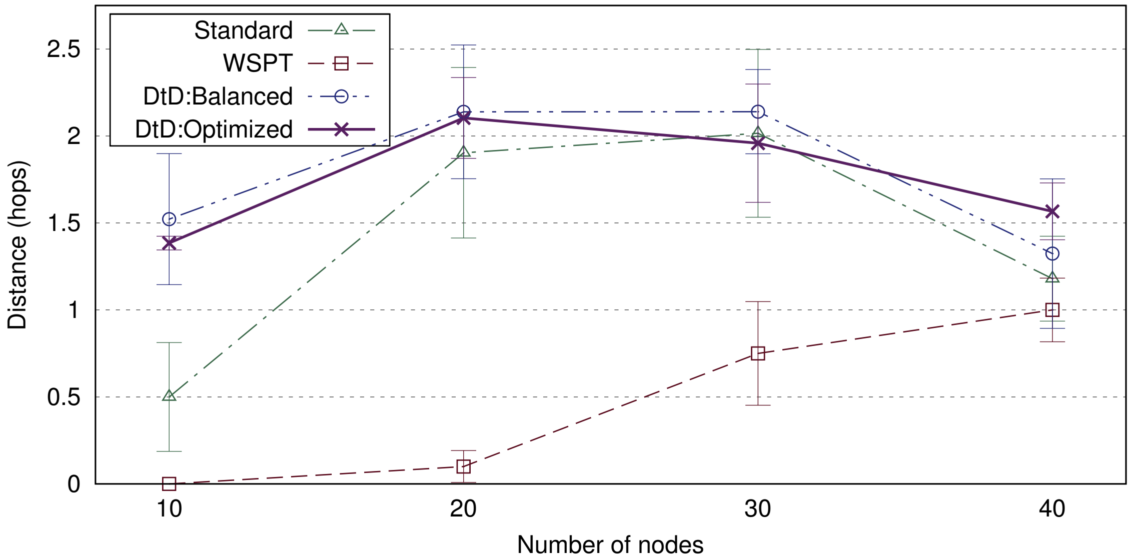

Figure D.13: Distance to stream source in mobile tour guide for the *Waypoint* mobility pattern

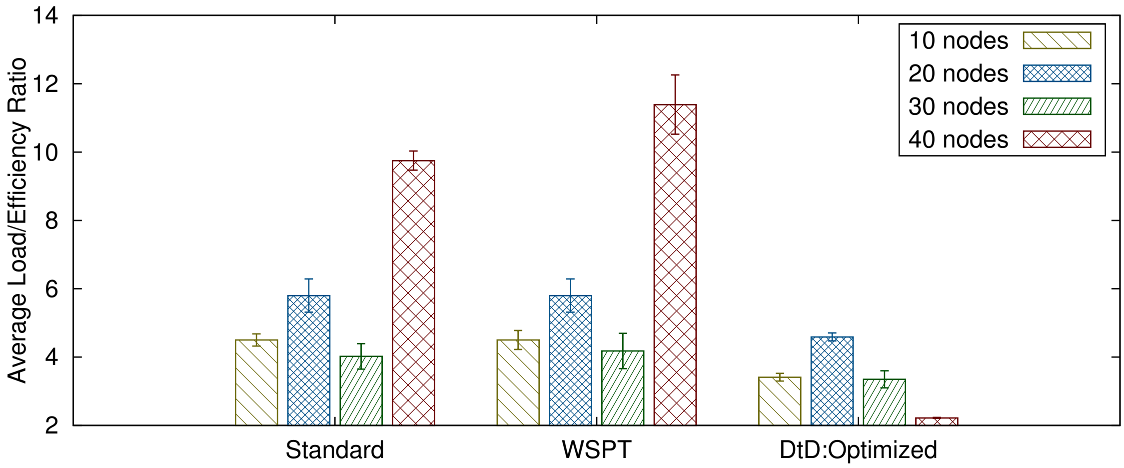

Figure D.14: Average load efficiency ratio in mobile tour guide for the *Waypoint* mobility pattern

each other, making it impossible to overcome the physical radio limitations.

Figures D.15 and D.16 represent the static pattern. In this type of environment, $\mathcal{DTD}$:Optimized largely improves the reception quality when compared to all other approaches, which tend to achieve similar results if the network conditions are the same over time. This improvement is achieved with the right balance between the weights that affect the associations performed between nodes. Since the set of association options is always the same in static scenarios, the improvement achieved in terms of reception quality requires longer paths to the information source node. However, the distance tends to be reduced as the networks grow, since the increase in the number of nodes implies a larger set of association options.

Regarding the average load efficiency ratio (Figure D.17), the $\mathcal{DTD}$:Optimized follows the same behavior as compared with scenarios with mobility, delivering more packets per

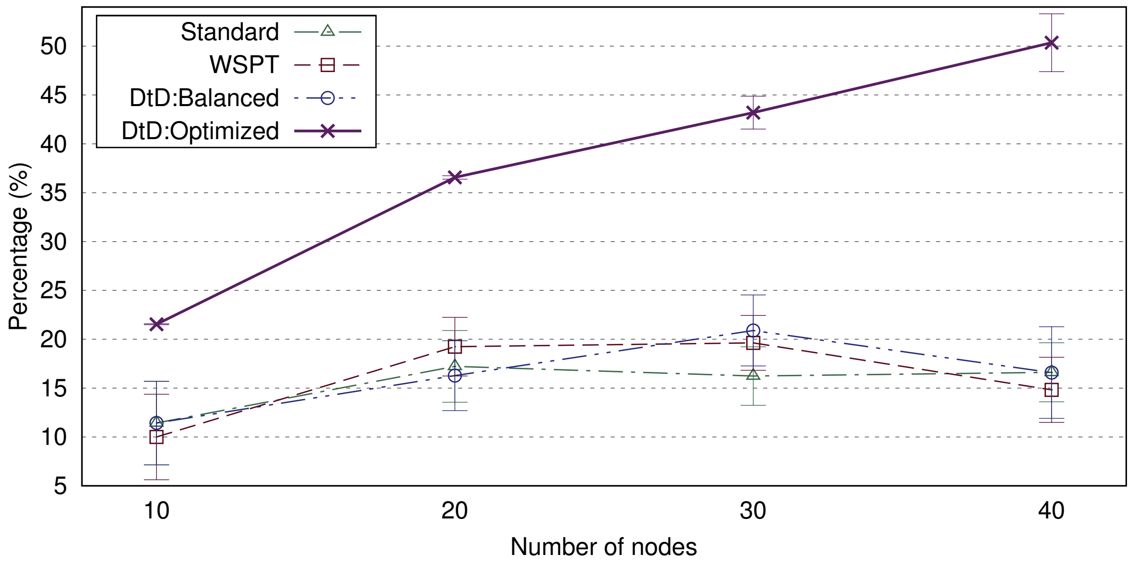

Figure D.15: Stream reception quality in a soccer stadium for the *Static* environment

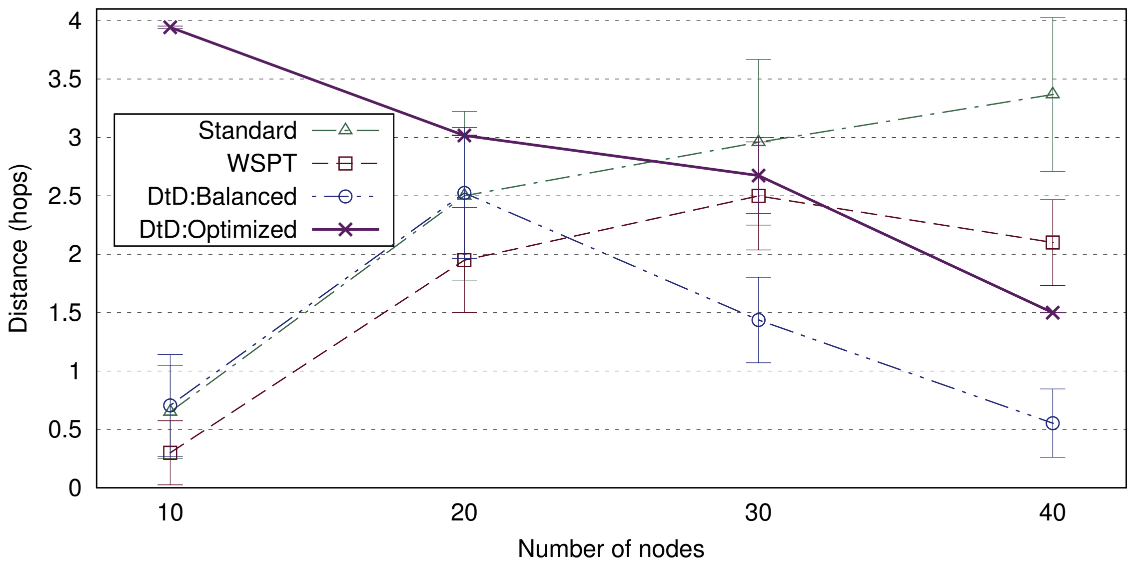

Figure D.16: Distance to the stream source in a soccer stadium for the *Static* environment

multimedia flow and shortening the distances when the nodes in the network increase.

Figures D.18 and D.19 represent the edge pattern. In this type of environment, *DTD*: Optimized widely improves the reception quality. The nodes are induced to reach the edge of the environment where they will remain until the end of a soccer match. This improvement is also attributed to the correct balance between the *DTD*:Optimized weights that affect the associations performed between nodes. Following the same behavior of the Figure D.17, the *DTD*:Optimized also improves the average load efficiency ratio according to the Figure D.20.

Table D.3 presents a summary of the reception quality percentual gain determined by the percentage of reception quality of stream values measured in the Figures D.9, D.12, D.15

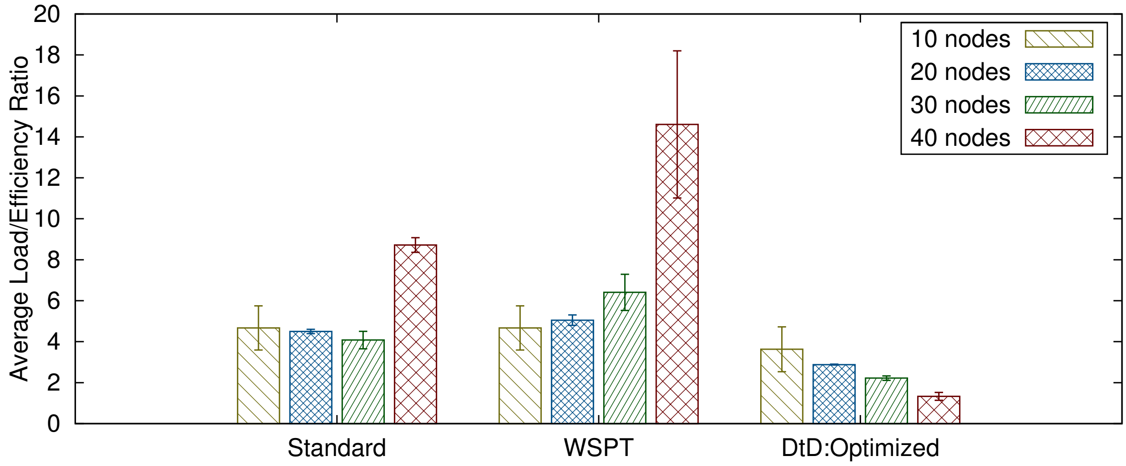

Figure D.17: Average load efficiency ratio in a soccer stadium for the *Static* environment

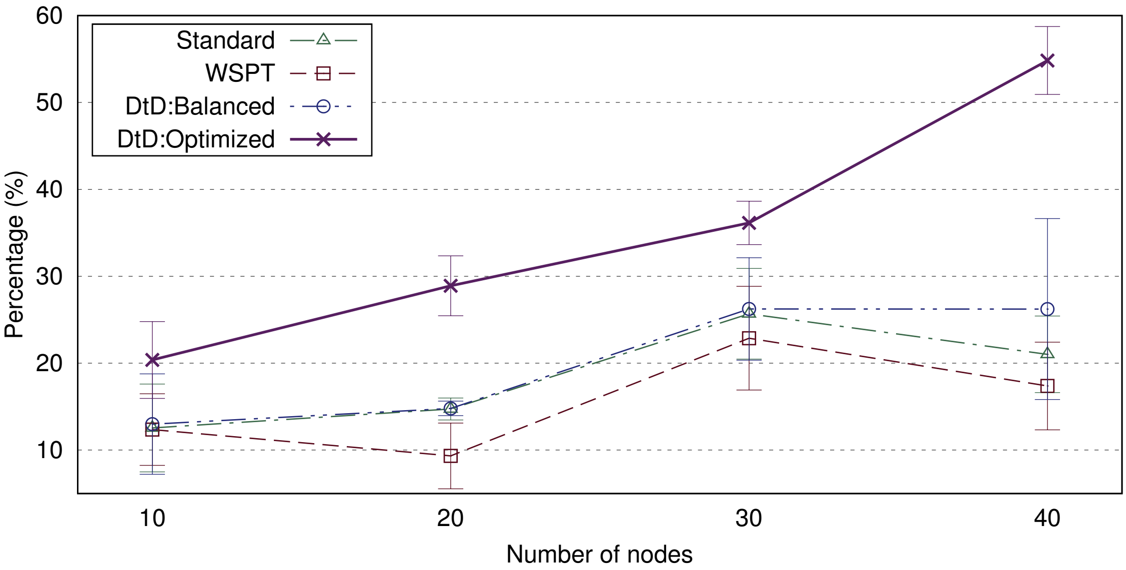

Figure D.18: Reception quality of stream in a soccer stadium for the *Edge* environment

and D.18, for both, mobile tour guide and soccer stadium scenarios. The gain is calculated using *Standard* as reference approach, and the results clearly demonstrate the benefit of the $\mathcal{DTD}$:Optimized in terms of the percentage of streaming multimedia reception quality. Evidently, the $\mathcal{DTD}$:Optimized outperforms all other mobility models considered, exhibiting superior efficiency in disseminating multimedia streaming over different scenarios and node variations. The highest benefit was noted for the case of the *Nomadic* mobility pattern over guide tour scenario. For the highest number of nodes considered in this scenario (40), our approach outperforms WSPT and $\mathcal{DTD}$:Balanced in terms of percentage point (pp) by 456pp and 354pp, respectively.

The presented results show the benefits provided by our Device-to-Device criteria in terms of (i) stream reception quality; (ii) distance to stream source; and (iii) average load efficiency ratio. In this article we analyzed the technical and social behaviors as complementary met-

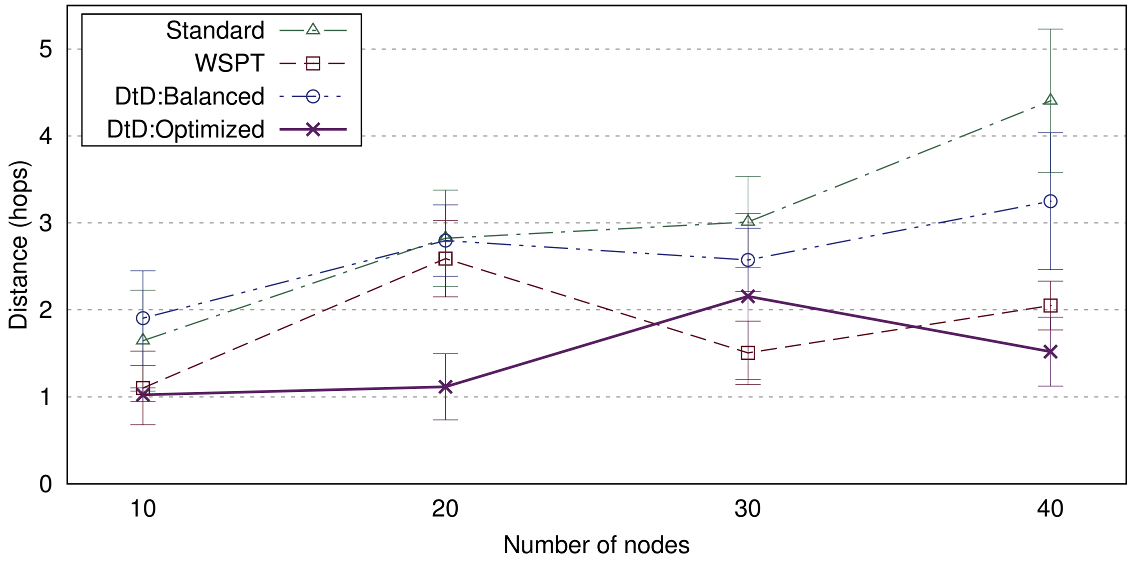

Figure D.19: Distance to stream source in a soccer stadium for the *Edge* enviroment

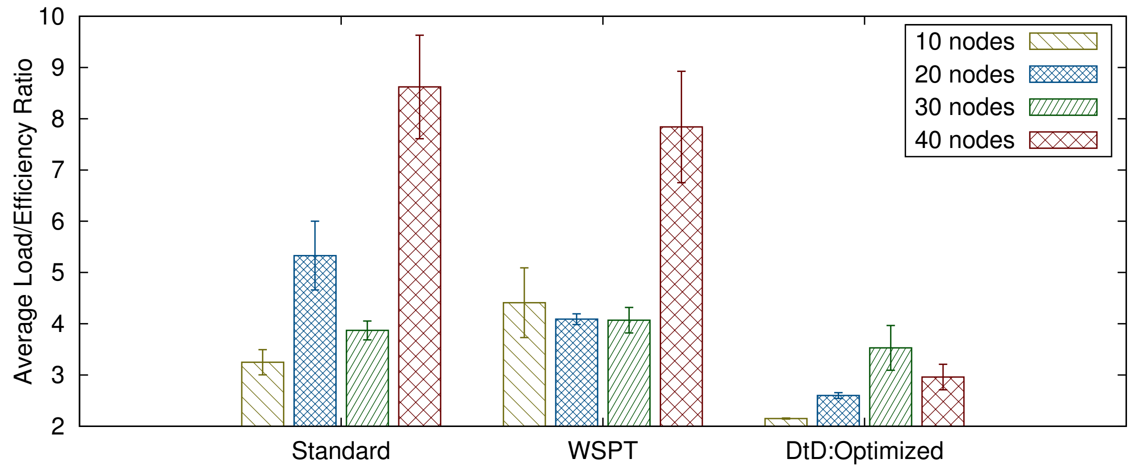

Figure D.20: Average load efficiency ratio in a soccer stadium environment for the *Edge* environment

rics to calculate the Device-to-Device criteria to nodes associations in order to support the dissemination of multimedia streaming. A study considering only technical behavior at MAC layer, as described in [Gea12a] and [Gea12b], contributed to improve the communities quality aggregations by reducing the network time, utilization and nodes reassociations. To disseminate multimedia contents, instead using only technical information at MAC layer, we consider necessary to evaluate the quality experienced by the interested users in the social network as well. Thus, adding the social network behavior experienced by the criterion $\mathcal{FF}_{i,j}$, the users that belongs to the same social groups will have higher probability to share the same content, enforcing the nodes affinity at the network level.

Table D.3: Reception quality percentual gain (using *Standard* as reference)

Nomadic	WSPT (%)	DtD:Balanced (%)	DtD:Optimized (%)
10 nodes	6.266	0.343	143.462
20 nodes	-28.950	5.171	153.934
30 nodes	-19.839	58.108	244.409
40 nodes	85.930	188.008	542.404
Waypoint	WSPT (%)	DtD:Balanced (%)	DtD:Optimized (%)
10 nodes	-0.826	14.131	61.367
20 nodes	9.980	39.122	98.589
30 nodes	5.131	57.747	133.854
40 nodes	1.501	103.607	230.100
Edge	WSPT (%)	DtD:Balanced (%)	DtD:Optimized (%)
10 nodes	-1.479	3.550	62.382
20 nodes	-36.687	0.445	96.187
30 nodes	-10.939	2.1479	40.712
40 nodes	-17.351	24.756	160.808
Static	WSPT (%)	DtD:Balanced (%)	DtD:Optimized (%)
10 nodes	-12.500	0.001	88.457
20 nodes	11.684	-5.478	25.060
30 nodes	20.936	28.747	166.100
40 nodes	-10.848	-0.173	202.917

D.6 Conclusion

This article proposed a novel approach to disseminate multimedia streaming considering Device-to-Device behaviors. The main contribution of this article considers the possibility to merge technical and social behaviors from the user's social networks accounts, over mobile and static networks, enhancing the dissemination of multimedia process. We have shown through two case studies how our proposed solution tends to be more efficient in terms of: reception quality of multimedia streams, distance to stream source, and average load ratio, according to the baselines analyzed. Finally, this article represents a first step in merging technical and social relationship interests in order to perform a complete Device-to-Device behavior into mobile infrastructureless network scenarios.

D.7 Acknowledgments

This work was supported by the Fundação para Ciência e Tecnologia - FCT, through the grant SFRH/BD/62511/2009. The authors also thank the support provided by Multiple Context-based Wireless Mesh Networks (MC-WMNs), FCT Project PTDC / EEA-TEL / 120176 / 2010.

References

[80297] I. 802.11. Lan/man wireless lans, June 1997. Accessed on the 12 of march 2007.

[DH07] E. M. Daly and M. Haahr. Social network analysis for routing in disconnected delay-tolerant manets. In *Proceedings of the 8th ACM international symposium* *on Mobile ad hoc networking and computing*, MobiHoc '07, pages 32–40, New York, NY, USA, 2007. ACM. http://doi.acm.org/10.1145/1288107.1288113.

[DXT09] T. Dinh, Y. Xuan, and M. Thai. Towards social-aware routing in dynamic communication networks. In *Performance Computing and Communications Conference (IPCCC), 2009 IEEE 28th International*, pages 161 –168, dec. 2009.

[FV11] F. Fabbri and R. Verdone. A sociability-based routing scheme for delay-tolerant networks. *EURASIP J. Wirel. Commun. Netw.*, 2011:1:1–1:13, January 2011. http://dx.doi.org/10.1155/2011/251408.

[Gea12a] L. Guardalben and et al. Improving mac layer association through social-based metrics in mobile networks. *IEEE Communications Magazine*, 50(6):91–98, 2012.

[Gea12b] L. Guardalben and et al. Self-organizing decentralized wireless management through social-based metrics. In *NOMS*, pages 1199–1202, 2012.

[GGSS13] T. Gomes, L. Guardalben, P. Salvador, and S. Sargento. A spanning tree protocol over mobile wireless ad hoc networks. In *IEEE International Conference on Communications 2013: IEEE ICC'13 - 3rd IEEE International Workshop on Smart Communication Protocols and Algorithms (SCPA 2013) (ICC'13 - IEEE ICC'13 - Workshop SCPA)*, pages 1109–1114, Budapest, Hungary, June 2013.

[GK12] M. K. Gulati and K. Kumar. Qos routing protocols for mobile ad hoc networks: a survey. *Int. J. Wire. Mob. Comput.*, 5(2):107–118, May 2012. http://dx.doi.org/10.1504/IJWMC.2012.046783.

[Gol89] D. E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1st edition, 1989.

[GSC+96] A.-V. T. W. Group, H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. Rtp: A transport protocol for real-time applications, 1996.

[HCY11] P. Hui, J. Crowcroft, and E. Yoneki. Bubble rap: Social-based forwarding in delay-tolerant networks. *Mobile Computing, IEEE Transactions on*, 10(11):1576 –1589, nov. 2011.

[IEE07] IEEE std. 802.11-2007, part 11: wireless LAN medium access control (MAC) and physical layer (PHY) specifications, June 2007.

[KSHC12] J.-L. Kuo, C.-H. Shih, C.-Y. Ho, and Y.-C. Chen. A cross-layer approach for real-time multimedia streaming on wireless peer-to-peer ad hoc network. *Ad Hoc Networks*, (0):–, 2012.

[LDS03] A. Lindgren, A. Doria, and O. Schelén. Probabilistic routing in intermittently connected networks. *SIGMOBILE Mob. Comput. Commun. Rev.*, 7:19–20, July 2003. http://doi.acm.org/10.1145/961268.961272.

[LKPG11] M. Lindeberg, S. Kristiansen, T. Plagemann, and V. Goebel. Challenges and techniques for video streaming over mobile ad hoc networks. *Multimedia Systems*, 17:51–82, 2011. 10.1007/s00530-010-0187-8.

[LN04] T. Liu and S. Nelakuditi. Disruption-tolerant content-aware video streaming. In *Proceedings of the 12th annual ACM international conference on Multimedia*, MULTIMEDIA '04, pages 420–423, New York, NY, USA, 2004. ACM. http://doi.acm.org/10.1145/1027527.1027627.

[MGKK12] S. Mantzouratos, G. Gardikis, H. Koumaras, and A. Kourtis. Survey of cross-layer proposals for video streaming over mobile ad hoc networks (manets). In *Telecommunications and Multimedia (TEMU), 2012 International Conference on*, pages 101 –106, 30 2012-aug. 1 2012.

[MMDA10] A. Mtibaa, M. May, C. Diot, and M. Ammar. Peoplerank: Social opportunistic forwarding. In *INFOCOM, 2010 Proceedings IEEE*, pages 1 –5, march 2010.

[NS-10] NS-3. Manual ns-3.9, 2010. http://www.nsnam.org/docs/release/3.9/manual.html.

[TLLL10] P. Tournoux, E. Lochin, J. Leguay, and J. Lacan. Robust streaming in delay tolerant networks. In *Communications (ICC), 2010 IEEE International Conference on*, pages 1 –5, may 2010.

[VWS11] A. Vetro, T. Wiegand, and G. Sullivan. Overview of the stereo and multiview video coding extensions of the h.264/mpeg-4 avc standard. *Proceedings of the IEEE*, 99(4):626 –642, april 2011.