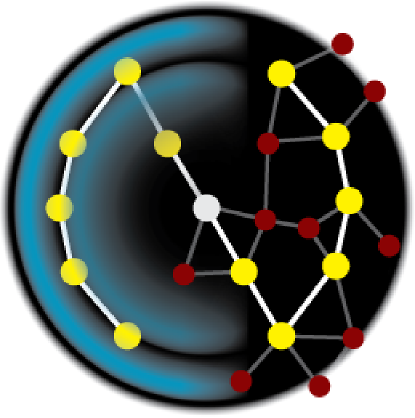
Delft University of Technology
Master's Thesis in Computer Engineering

Offloading Mobile Data Traffic to Wi-Fi Through MASS-ANET: Mobile Assisted Ad-hoc Networking

Marco Cova



Offloading Mobile Data Traffic to Wi-Fi Through MASS-ANET: Mobile Assisted Ad-hoc Networking

Master's Thesis in Computer Engineering

Embedded Software Group
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology
Mekelweg 4, 2628 CD Delft, The Netherlands

Marco Cova

4th October 2011

Author

Marco Cova

Title

Offloading Mobile Data Traffic to Wi-Fi Through MASS-ANET:
Mobile Assisted Ad-hoc Networking

MSc presentation

11th October 2011

Graduation Committee

Prof.dr. Koen Langendoen	Delft University of Technology
Dr. Stefan Dulman	Delft University of Technology
Dr.ir. Anthony Lo	Delft University of Technology
Ir. Niels Brouwers	Delft University of Technology

Abstract

Massive sales of smartphones and tablets have enabled more people to access the internet with a 3G connection than ever before. Predictions indicate an exponential growth of mobile data traffic world-wide reaching 6.3 exabytes per month by 2015, a 26-fold increase over 2010 [9]. Unfortunately, the current infrastructure will not be able to meet these demands. Already now, a speed reduction or even lack of connection can be noticed in crowded places. Service providers are addressing the problem by deploying Femtocells and Wi-Fi Access Points (APs) around cities. However, due to their limited connection range, the projected redirected traffic, including free APs at home and office, is expected to be only 39% by 2015 [9].

The intention of this thesis is to extend the range of Wi-Fi APs as internet gateways through multi hopping. Furthermore, phone calls and data exchange between devices in proximity are supported and also offloaded from the 3G data network. Therefore, we offload part of the traffic from the mobile data network to Wi-Fi. An added benefit is a reduction in energy consumption. However, current Mobile Ad-hoc Network (MANET) techniques are unable to scale or to handle high mobility. Therefore, we propose a novel approach called "Mobile Assisted Ad-hoc Networking". MASS-ANET can be applied to any kind of mobile ad-hoc network where nodes can use both a global and a local transmission link. In this project, Wi-Fi local connectivity is used to create a mobile ad-hoc network. 3G links act as a global coordination medium with the help of a central server.

The feasibility of the MASS-ANET is determined with the help of simulations. Subsequently modern commodity smartphones are used as prototypes in validation experiments. The proposed solution is verified using a real-world application: exchanging real-time voice traffic over several hops in a mobile ad-hoc Wi-Fi network.

Preface

Three years ago, I would have never thought about going abroad to continue my university studies. Only during the last year of my bachelors I started thinking about that. I was attracted to the master program offered by the Delft University of Technology because it focuses both on software and hardware. After two years of hard but rewarding work, now I'm here defending my master thesis.

My thesis subject underwent an evolution through four different topics up to the final one. Anyway, I was pleased because it turned out to be an amazing adventure and I learned a lot. Especially, as I had the chance to combine both theoretical and practical experiences in only one project.

However, it would not have been possible if it had not been for my parents. I would like to thank them for the incredible chance they gave me and for the support during these two years.

I would like to thank professor Koen Langendoen for pushing me in the right direction and giving me hints to improve my work and my English.

I would like to thank my supervisor Niels Brouwers because he was always available for me, and in the difficult moments I always got out from his office with a smile on my face.

Thanks to all the PhDs of the Embedded Software Group, among them Venkat and Andreas, as well as Nicola, Nassos and Kristian and all the friends who helped me with the experiments.

Thanks to Szymon Jakubczak for the open-source "Barnacle Wi-Fi Tether" application, a key tool for my thesis prototype.

Besides the thesis work, a special thank to Désirée and all my friends who made me feel at home even if in a foreign country.

Delft, The Netherlands
4th October 2011

Contents

Chapter 1

Introduction

Nowadays mobile devices with internet connectivity are extensively used both for business and entertainment. They are even considered a “status symbol” and a “must have”. This is reflected by the huge sales and the stiff competition between hardware manufacturers to continuously come up with more powerful and attractive devices.

The aggregate mobile traffic in 2010 reached the considerable volume of 237 petabytes per month [9]. The current infrastructure is already struggling with the load, yet the global mobile internet traffic is projected to grow by a factor of 26 in the next five years to staggering 6.3 exabytes per month in 2015. 66.4% of the traffic will be due to video [9]. The current 3G infrastructure will simply not be able to scale up to this demand.

Telecommunication companies are aware of the situation. To increase the network capacity providers start to deploy femtocells and, in the immediate future, Wi-Fi Access Points (APs). Femtocells are small cellular base stations that support up to 16 mobile connections. Wi-Fi APs are useful to enable seamless handover of internet data connections between 3G and Wi-Fi [43, 53].

Although offloading to Wi-Fi is cheap, there are limitations. The projected total offload traffic for smarphones and tablets will be only 39% in 2015 [9], see Figure 1.1. The range of a typical Wi-Fi connection is only about 100m (200m for the IEEE 802.11n standard) [1, 17, 23], which severely restricts its beneficial effect. Innovative solutions are required to increase the percentage of offloaded traffic.

We researched new techniques to increase the 3G network capacity without the need for changing the current infrastructure. The goal of this thesis is to extend the range of Wi-Fi APs by routing information through intermediate devices in order to offload 3G data traffic to Wi-Fi. Therefore, the devices need to be connected to each other through a Wi-Fi ad-hoc network.

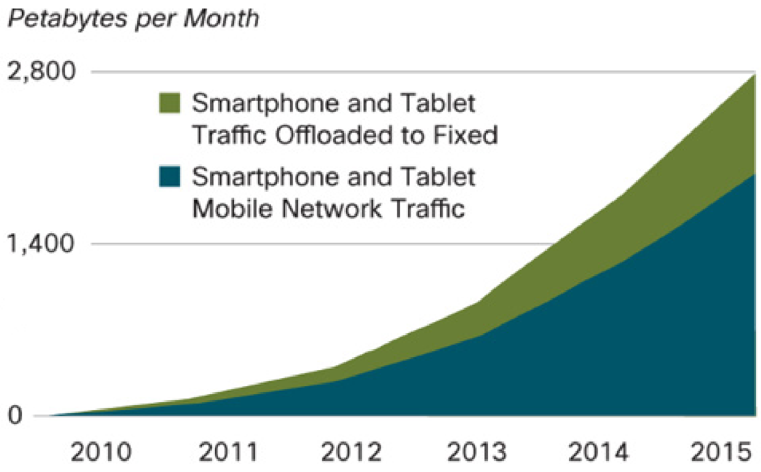


Figure 1.1: 39% of mobile data traffic will be offloaded by 2015 [9].

Mobile Ad-hoc Networks (MANET) are notoriously difficult to maintain due to their mobility and scalability issues. We decided to investigate new alternatives because current MANET techniques such as reactive and proactive protocols are unfeasible: the former due to a high message complexity (see Section 2.1.1) and the latter due to scalability issues, especially in the context of mobile networks (see Section 2.1.2).

We propose a novel approach called "Mobile Assisted Ad-hoc Networking" (MASS-ANET) that exploits a local and a global channel. Our project uses the 3G mobile internet (global channel) as a coordination medium for a Wi-Fi (local channel) ad-hoc network. The 3G connection is linked to a central server in the cloud. The server has a global view of the Wi-Fi network and assists with the node discovery and routing functions. This dual-channel solution merges the best of both worlds. The bulk of the data is transferred through Wi-Fi whenever possible, which is cheaper in terms of both energy consumption and money, and 3G allows for an easy and instantaneous connection to a central server.

Geographical routing was taken into account in order to route inside the ad-hoc Wi-Fi network. However, after initial tests it became clear that this solution comes at a high price. The GPS chip drains the smartphone batteries at a fast pace. Moreover, it does not provide indoor coverage. In [35] the measured GPS average power consumption for the Nokia N95 is 324mW and in [33] the HTC Dream GPS is reported to consume between 400mW, in sleep mode, and 600/800mW, when active, with peaks of 1400mW. Especially in a mobile environment, energy conservation should be a fundamental characteristic of any proposed algorithm.

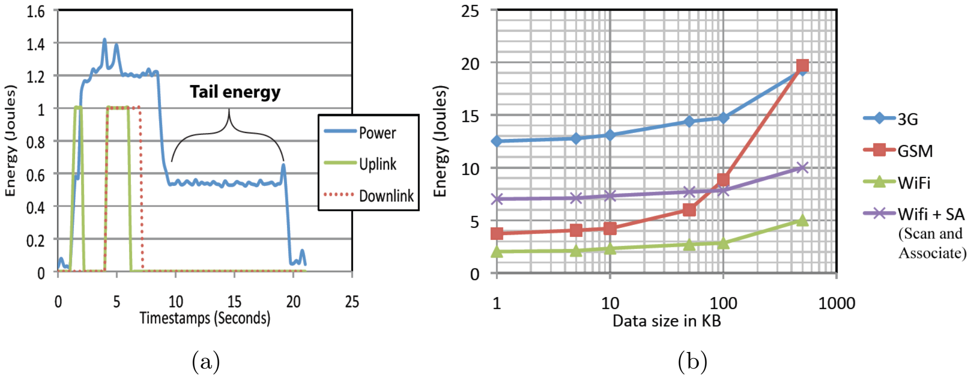


Figure 1.2: (a) 3G: Power Profile - 50KB and (b) Average energy consumption comparison for different download data size [39].

In a first "naive" approach the 3G link with the central server is enabled in each node. Therefore, the server has the complete global view of the Wi-Fi ad-hoc network and it can easily control the routing paths. However, this approach consume a lot of energy.

Although it may seem that the energy consumption of a radio chip is proportional to the amount of data transferred through it, this assumption does not hold for 3G. As seen in Figure 1.2(a), most of the power consumption of a 3G chip is due to tail energy that lasts up to 12.5 seconds [39]. The tail energy is the energy consumed by the radio chip after the actual data transmission, when the radio is in stand-by mode. Therefore, 3G has a high consumption even if the packets sent are small because periodic transmissions prevent the radio from going into sleep mode. On the contrary, Wi-Fi is not affected by tail energy and its efficiency grows dramatically with increasing data sizes Figure 1.2(b).

The previous considerations lead to the exploration of multiple solutions that reduce the number of devices having to constantly communicate to the central server over 3G. The decrease of 3G active devices was achieved through a graph structure called a *Connected Dominating Set* (CDS) [15]. The CDS is formed by a subset of the network, and only part of the nodes belonging to the CDS needs to actively interact with the server. The CDS algorithm used is a variation of [29].

We used the discrete time simulator NetLogo [48] in order to investigate the behaviour of MASS-ANET in distinct application scenarios, such as different network densities and mobility patterns. We used commodity devices as prototypes to evaluate MASS-ANET performances in real-life situations. The maintenance phase was investigated in both static and mobile networks. A real-time voice communication was exchanged between devices not in range as proof of concept.

The remainder of this thesis is structured as follows: related work on mobile ad-hoc networks, ad-hoc Wi-Fi, discovery, control, routing protocols and CDS algorithms are presented in Section 2. The design with both a naive approach and the CDS algorithm enhancement is presented in Section 3, followed by the algorithm's analysis in Section 4. Section 5 presents the prototype implementation and Section 6 results from real-world experiments. Conclusions and future work are presented in Section 7.

Chapter 2

BACKGROUND

Different technologies and techniques were considered during the development of MASS-ANET. This chapter gives an insight into the current knowledge in the mobile network field and the required back-ground.

In Section 2.1, an introduction of mobile ad-hoc networks and several protocols are presented. Section 2.2 gives an overview on the most famous wireless technology with a special paragraph in Section 2.2.3 on ad-hoc Wi-Fi mode. Section 2.3 provides a small survey on Connected Dominating Set algorithms. Finally, Section 2.4 discusses some related projects.

2.1 Mobile Ad-hoc Network (MANET)

A Mobile Ad-hoc Network is an autonomous system of mobile devices connected through wireless ad-hoc links. Besides the source and sink functions, each element of a MANET must be able to forward data packets on a correct route. This is a difficult task because, as the name suggests, the network is mobile. The network topology is highly affected by the mobility of the devices and the path between two given nodes can change over time. Therefore, network protocols deal with two problems: routing (i.e., path discovery) and node discovery. Two categories of protocols called *reactive* (i.e., on-demand) and *pro-active* (i.e., table-driven), described respectively in Section 2.1.1 and 2.1.2, provide techniques to solve both problems. Other protocols only provide specific functionalities. In Section 2.1.3 and 2.1.4 two powerful protocols are described to solve the routing problem. However, additional services are needed to translate a node ID into its address (i.e., node discovery). Distributed Hash Table (DHT) is a possible solution that can be adopted at the bottom of the routing protocol [47]. Through the DHT technique the responsibility of maintaining the mapping between node IDs and addresses is distributed among a subset of nodes. Therefore, the network can scale and better handle network changes compared to a centralized approach.

2.1.1 Reactive Protocols

The family of reactive protocols, which includes the famous “Ad-hoc On-demand Distance Vector” (AODV) [8] and its successor “DYnamic Manet On-demand routing” (DYMO) [21], tries to find nodes and routes on demand by flooding the network with “Route Request” (RREQ) messages. Each RREQ message keeps a list of visited nodes towards the destination. Intermediate nodes save the visited node list of RREQ messages as a path to the origin of the communication. When the RREQ reaches the destination, a “Routing Reply” (RREP) message is sent back to the source using the same path saved during the first way and the communication can start.

The disadvantage of these kinds of protocols is a high message complexity that causes congestion in large and mobile networks.

2.1.2 Pro-active Protocols

Proactive protocols maintain knowledge of all the nodes and routes of the network in each node. Therefore, both the node discovery and the routing problems are solved. There are two major classes. In the “Link State Routing” (LSR), every node holds a complete map of the network which is constantly updated. In the “Distance Vector Routing” (DVR) class, the nodes record the direction to which to forward the packet and the distance from its destination.

The most recent and fastest protocols of this family are “BABEL” [22], a DVR protocol, and two LSR: “Optimised Link State Routing” (OLSR) [49] and “Better Approach To Mobile Ad-hoc Network” (B.A.T. M.A.N.) [3, 37]. Several papers [12, 34] investigate the performance of these protocols on different aspects. The conclusions of [34] show B.A.T.M.A.N. as the winner in stability and packet delivery, while BABEL offers the highest multi-hop bandwidth and the fastest route repair time. The tests also highlight the slow convergence times of these protocols that make them not suitable for mobile networks but, only for quasi-static ones. Scalability is also an issue with this family of protocols.

2.1.3 Geographic Routing

Geographic routing is the first of the two protocols described in this chapter that provides only routing functionalities. Geographic routing forwards packets from hop to hop based on the current packet position, destination position and neighbour positions [20].

Initial geographic routing techniques were completely stateless and they used a greedy approach to route a packet to the closest neighbour to the final destination. However, this method presents problems in case of local minimum situations. In fact, the closest node to the destination is not always connected to another node in the direction of the destination. Therefore, other strategies, as the "Face Routing" algorithm [14, 40] were developed to recover the packet from dead end paths.

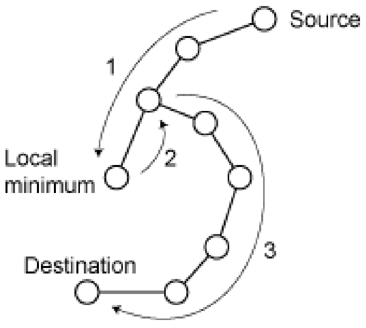


Figure 2.1: Geo-routing local minimum example (1) with packet recovery (2-3).

Even if geographic routing protocols can provide a good solution, they need a localization strategy. An example is an active GPS receiver but, as already explained in the introduction, it consumes too much energy. Furthermore, GPS is not available in-doors. Moreover, the node discovery functionality is not provided and it needs an additional strategy.

2.1.4 Beacon Vector Routing

Beacon Vector Routing (BVR) is an algorithm very similar to geographic routing. BVR is also built on greedy forwarding of packets based on coordinates of nodes. However, the coordinates are not geographical. In fact, they are assigned counting the hop distance from a non-mobile subset of nodes in the network elected as "Beacon".

This protocol is very powerful. However, in case of high mobility, if the hop count distance between beacons is too high, the nodes cannot keep their coordinates information up-to-date. The BVR protocol, as the geographic one, does not provide node discovery.

2.2 Wireless Technologies Overview

The network protocols described so far, as our network, need a transmission media. The mobile network presented in this thesis uses two kinds of wireless technology to connect all the nodes together and to allow a connection to a central server. An overview of the main two categories is presented in Section 2.2.1 and Section 2.2.2.

Standard		**DownLink**	**UpLink**	**Range**
		Theroetical Mbps (Avarage Mbps)		Km
2G	Global System for Mobile Communication (GSM)	0.009 (-)	0.009 (-)	25
2.5G	General Packet Radio Service (GPRS)	0.111 (0.046)	0.019 (0.014)	25
2.75G	Enhanced Data rates for GSM Evolution (EDGE)	0.375 (0.17)	0.058 (0.029)	25
3G	Universal Mobile Telecommunications System (UMTS)	0.375 (0.22)	0.065 (0.029)	29
	UMTS using Wideband CDMA (W-CDMA)	2 (0.781)	0.149 (0.058)	29
Pre-4G	High Speed Packet Access (HSPA 14)	14.4 (2)	5.76 (0.683)	29
	Evolved High-Speed Packet Access (HSPA+)	56 (-)	22 (-)	29
	Worldwide Interoperability for Microwave Access (WiMAX)	100 (3-6)	56 (1)	6.5
	Long Term Evolution (LTE)	100 (5-12)	50 (2-5)	6.5/30
4G	Worldwide Interoperability for Microwave Access 2 (WiMAX 2)	100mobile 1024fixed (-)	60 (-)	6.5
	Long Term Evolution Advanced (LTE Advanced)	100mobile 1024fixed (-)	- (-)	6.5/30

Table 2.1: Licensed frequency technologies overview, approximate data [11]

2.2.1 Licensed Frequencies

Licensed frequencies are a part of the frequency spectrum that may be used only by the owner, usually a telecommunication operator, that bought the permission to use that frequency from the government. These frequencies are the ones we usually use when we want to make a phone call or check the mail with our smartphone. In this research we used standard 3G, available in current smartphones, in order to communicate with the central server and to coordinate the network. Table 2.2 presents an overview of range and speed information, of the past, present and future licensed frequency technologies. The values are approximate. They depend on several elements such as distance or obstacles (e.g., in case of indoor operations). In fact, WiMAX can operate with a maximum range of 50Km with a line of sight connection, but the resulting bit-rate would be much lower compared to a shorter distance transmission due to a high bit error rate. The bandwidth is also affected by congestion due to a high number of connections to the same transmitting tower. Therefore, in a highly populated area, such as a city, the transmitting range must be reduced, otherwise the bandwidth could be compromised.

2.2.2 Unlicensed Frequencies

Regulation free spectrum frequencies are called unlicensed and can be used without constraints except for the transmitting power. An example is a Wi-Fi 802.11 access point or a bluetooth link that everyone can use at home. Furthermore, a Wi-Fi 802.11b/g card is also provided in every smartphone nowadays on the market, such as the ones we used as prototypes. The most expensive devices are even supplied with the latest 802.11n standard. Unfortunately the specifications of the Wi-Fi ad-hoc mode bit-rate only support up to 11Mbps. During our research, we observed a maximum bandwidth just below 1Mbps, at a distance of 100m. These measurements confirm the results reported in [17]. Table 2.2.2 shows a small overview of the most used free wireless technologies.

Standard	**DataLink** Mbps	**Range** m (in/outdoor)
Wi-Fi 802.11b	11	38/140
Wi-Fi 802.11g	54	38/140
Wi-Fi 802.11n	150 (300)	70/250
ZigBee based on 802.15.4	0.244	10/75
Wireless USB	53 - 480	3 - 10 indoor
Near Field Comm. NFC	0.414	0.1
Bluetooth	3	10/100 (class dependent)
Infrared	4	5 (line of sight)

Table 2.2: Unlicensed frequency technologies overview, approximate data [1, 17, 23].

2.2.3 Wi-Fi Ad-Hoc Mode

The Wi-Fi is one of the two communication channels adopted in MASS-ANET. The most common Wi-Fi connectivity mode is “Infrastructure mode”. We use the Wi-Fi infrastructure mode for example at home or at the office, when we connect our device to an access point, router or switch, to enter the home/office network or internet. The access point is usually connected to the internet through a wired connection.

In order to build a decentralized peer-to-peer device network we used a special Wi-Fi mode called “Ad-hoc mode”. Ad-hoc mode allows peer-to-peer communication between devices that are in range. This mode does not need any central point or fixed infrastructure. Two devices, in order to be able to “see” each other, only need to set the same Service Set Identifier (SSID) and channel.

2.3 Connected Dominating Set

On top of the Wi-Fi peer-to-peer network we use a particular graph structure called *Connected Dominating Set* (CDS). In this section we present a formal definition and an overview of the related work in this field.

The original definition of a Connected Dominating Set (CDS) dates back to 1979 [15]. Two properties hold in a CDS:

- A Dominating Set of any graph $G = (V, E)$ is a subset $C \subseteq V$ such that every vertex $v \in V$ either is in the set C or it is adjacent to a vertex in C
- C is a Connected Dominating Set if C is a Dominating Set and the subgraph induced by C is connected.

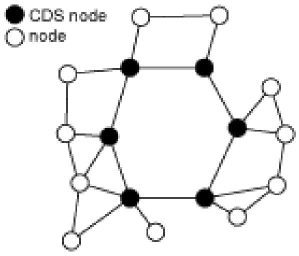


Figure 2.2: The CDS in this example is also Minimum (MCDS) - only black nodes are part of the CDS.

The first research efforts on the CDS topic were focused on the investigation of optimal algorithms to build Minimum CDS (MCDS), which was proved to be NP-Hard [6]. Only in the last decades the mobility and the maintenance of the network were taken into consideration. However, high mobility is still an open issue.

2.3.1 Maximal Independent Set

A CDS can be constructed using several techniques as presented in the next section of related work. However, in this section we introduce another graph structure that is the core of the CDS approach used in MASS-ANET.

An *Independent Set* in a graph is a set of vertices, no two of which are adjacent. A *Maximal Independent Set* (MIS) is an independent set that is not properly contained in any independent set [44]. A CDS can be build adding specific vertices to the MIS set.

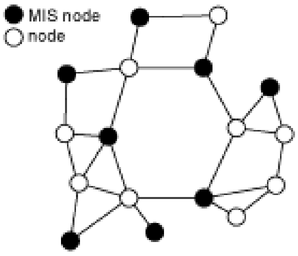


Figure 2.3: Maximal Independent Set example, only black nodes are part of the MIS.

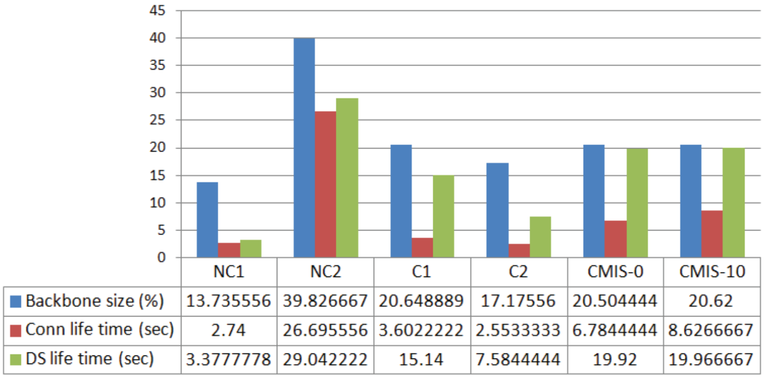
2.3.2 Related Work

We now present state-of-the-art CDS algorithms, some of which inspired the MASS-ANET design.

In [51] the authors propose a CDS algorithm composed of three phases: Backbone Selection Process, Backbone Connection Process and Backbone Maintenance Process. The algorithm is based on different control messages broadcasted every second to 1-hop neighbours, and every 10 seconds to 3-hop neighbours. The CDS is built incrementally. This protocol does not intrinsically handle high mobility. They faced the mobility problem checking the concentration of CDS nodes in a neighbourhood. If it is above a threshold, a CDS node is elected among them to remain in the CDS while the others change state leaving the CDS. The authors validate their idea only by simulation in an ideal world without mentioning the number of messages exchanged in the bootstrap phase or during mobility. They use different control messages and since one of them is broadcasted every 10 seconds, we assume that even in an ideal world this algorithm takes more than 10 seconds to build the CDS.

[38] proposed a new approach based on finding and interconnecting a Maximal Independent Set, which is a dominating set, with a minimum spanning tree (MST) from an initiator. The two phases, MIS discovery and interconnection, are interleaved in order to reduce the construction cost in terms of time and messages. Their simulations are very interesting because they implement different CDS protocols [4, 27, 42]. They simulated a total of 150 seconds in a mobile environment measuring the backbone size and life

time, see Figure 2.4. The best one among them reaches only a lifetime of 29 seconds.



	NC1	NC2	C1	C2	CMIS-0	CMIS-10
🟦 Backbone size (%)	13.735556	39.826667	20.648889	17.17556	20.504444	20.62
🟥 Conn life time (sec)	2.74	26.695556	3.6022222	2.5533333	6.7844444	8.6266667
🟩 DS life time (sec)	3.3777778	29.042222	15.14	7.5844444	19.92	19.966667

Figure 2.4: Comparison of simulation results: backbone size and life time [38].

The algorithm of Das et al. [4] finds a small dominating set S where the edges form a spanning forest of the graph. A minimum spanning tree (MST) is used to connect all the fragments from the first stage selecting the nodes, one by one, which covers most of them at 1- or 2-hop not in S.

Wu and Li [27] compute a CDS by subtraction. 2-hop neighbours are exchanged. If a node finds a direct link between any pair of its neighbours with higher IDs, it removes itself from the CDS. Additional rules are applied to further reduce the size of the CDS. Dai's protocol [16] extend the pruning rules of Wu.

Wan [42], similar to [38], proposes to find the maximal independent set by starting from an initiator node and using a spanning tree to traverse the graph. Then the MIS is connected together based on the hop counts from the root to form the CDS.

Zhou et al. [13] proposed a solution where only 1-hop information is needed. A CDS tree is generated from a unique leader distributively elected. Zhou's protocol assumes that the neighbourhood is known. The generation is based on a defer timer that is set, for each node, based on the number of uncovered neighbours. The probability of a node to be in the CDS is proportional to the number of its uncovered neighbours.

Three papers from Sakai et al. are presented in [30, 31, 32],which are the evolution of the same protocol. After a multi-initiator election phase, each initiator builds a dominated tree. When two leaf nodes are neighbours they can become bridges to connect the respective trees after having exchanged specific control messages with the root of their tree (their dominator). Fig-

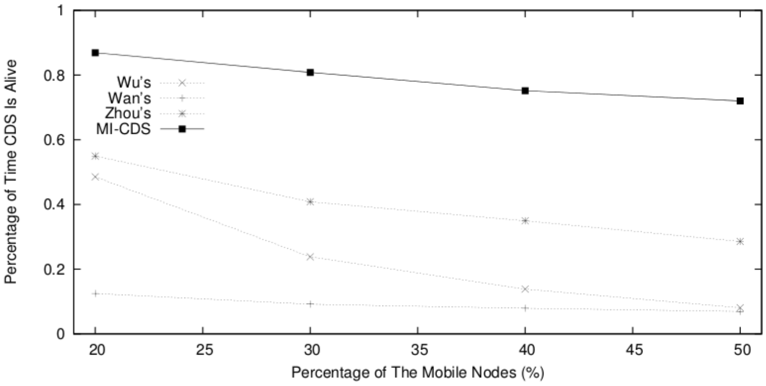


Figure 2.5: Percentage of time CDS is alive [30].

ure 2.5 illustrates the percentage of time the CDS is alive with respect to the percentage of mobile nodes between their Multi-Initiator protocols and three other protocols: [13, 27, 42]. Figure 2.6 shows a similar comparison between Dai's protocol, the Single-Initiator protocol [13] and their Multi-Initiator protocol with and without Extended Mobility Handling. The best CDS protocols, according to these figures, are the MI w/EMH that is alive for about 80% of the time with the 50% of mobile nodes and the Dai's protocol that is alive about the 61% of the time with nodes mobility of 80%.

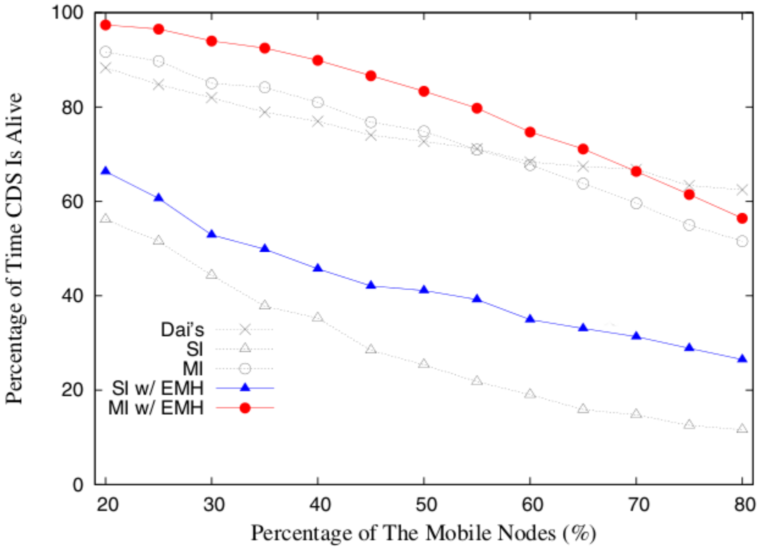


Figure 2.6: Percentage of time CDS is alive [32].

The last heuristic reviewed is proposed by Alzoubi et al. [29]. Alzoubi's protocol takes advantage of the Maximal Independent Set (MIS) properties to construct and maintain the CDS. Nodes belonging to the MIS are called *dominators*. They are separated at least by 2-hops. Each dominator is connected to all the other ones within a 3-hop distance through *connector* nodes. The maintaining phase approach of this algorithm is to maintain the MIS and its node connectivity. All the other nodes that are not dominators or connectors are called *dominatees*. At the beginning of the set-up phase all the nodes are in an initial state called *candidates*. A candidate node collects information about neighbourhood node IDs. The node with the lowest ID has the priority to change its state. All the other nodes wait and change state according to their smaller ID node neighbours. Even in the maintenance phase, if part of the network, composed by dominatee nodes, requires a new dominator, messages are exchanged to elect the node among them with the smallest ID. Therefore, the algorithm has linear time complexity. Due to its MIS feature and the relatively easy maintenance phase, we adopted and slightly modified this protocol for using it in MASS-ANET.

2.4 Android-based ad-hoc Wi-Fi projects

This section gives an overview of the current Android-based projects which exploit Mobile Ad-hoc Networks.

Dythr [46] and *Auto-BAHN* [50] projects use the Wi-Fi ad-hoc mode to disseminate short messages to all the smartphones in the neighbourhood. Dythr includes the messages in the SSID, or the network name. The messages are spread into the network through the Wi-Fi beacons. Each owner can choose to forward the message again or not. Auto-BAHN, instead, uses the classic approach of flooding the network. In both projects there is completely no knowledge of the network topology.

The *Serval* project [41] is based on ad-hoc Wi-Fi networks too. Serval is much more similar to our research with respect to the previous two projects because it builds and maintains a completely functional ad-hoc Wi-Fi network. Its goal is to provide phone call capabilities when part of the network is not covered by the telecommunication infrastructure, as during natural disasters. Phone calls can be made also inside the local MANET, in case the infrastructure is completely absent. However, Serval uses the network protocol OLSR that, as explained in Section 2.1.2, it is not really scalable and not suitable for mobile networks.

Chapter 3

DESIGN

In order to increase the effective range of Wi-Fi Access Points (APs), the devices need to be able to forward messages to one another along the correct route. Therefore, the devices need to be connected together in a Wi-Fi mobile ad-hoc network (MANET). A MANET handles multi-hop data transfer between nodes, which can be either smartphones or APs. In our proof of concept we use a smartphone as ending point of an end-to-end connection instead of an AP. We provide smartphone-to-smartphone real-time data transfer or phone calls, at several hops of distance through the Wi-Fi network.

Current MANET protocols are not proven to be ready to handle high mobility or dense networks as reviewed in Chapter 2. The goal of this chapter is to give an insight into the design of a new MANET technique, which we call “Mobile Assisted Ad-hoc Networking” (MASS-ANET). MASS-ANET is based on the assumption that, in addition to a short range radio link, the nodes in the network have a global link.

First considerations are discussed in Section 3.1. A “naive” approach is described in Section 3.2 while a more efficient technique, using a CDS algorithm, is presented in Section 3.3.

3.1 Design considerations

MASS-ANET assumes a global and a local communication link. In order to validate our research we used commodity smartphone devices as prototypes. They support 3G radio technology, which serves as the global data channel, and the Wi-Fi is used for local communication.

There are two features playing a key role in the design:

- The techniques for discovery, control and routing functions need to be fast, in terms of convergence time in order to deal with mobility.

- The energy consumption needs to be minimized. The 3G channel is expensive in terms of both money and energy consumption as explained in Chapter 1. Its consumption, at the network level, depends mostly on the actual number of nodes that exchange messages rather than the traffic volume to the 3G radio tail energy. Therefore, our primary goal is to reduce the power consumption by minimizing the number of nodes using the 3G link.

3.2 Naive Approach

Two services are essentials in a network: discovery and routing. The discovery function manages the knowledge of the network among nodes. The task of routing is to correctly deliver messages inside the network.

In many networks, including ours, nodes periodically broadcast beacon or "Discovery" messages to notify their presence to the neighbourhood. With the beaconing technique, each node only knows about its direct neighbours, but nothing about other nodes nor how to route towards them. Device discovery and routing functions are still missing.

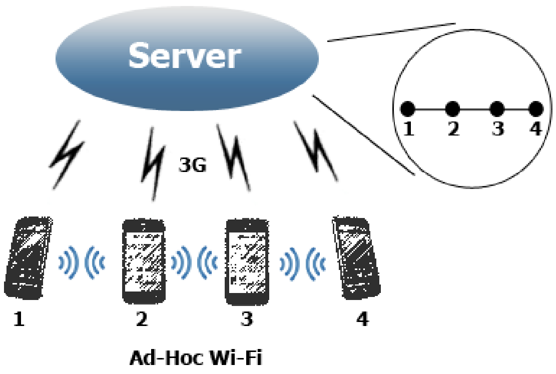


Figure 3.1: Naive approach: all smartphones are connected through 3G to the central server.

3.2.1 Discovery and Control

The simplest, naive approach to add discovery and control functionality is to implement a central server that each device can contact through the 3G connection. Each node constantly keeps the server up-to-date regarding its neighbour list on the Wi-Fi channel. Therefore, the server has always a complete view of the ad-hoc Wi-Fi network, as shown in Figure 3.1.

The discovery information can be exploited by the server, which has perfect knowledge of the network. The server can share its knowledge with nodes when needed.

3.2.2 Routing

In order to route a packet, the source node may request the server to set-up a path to the destination node. Subsequently, the server computes and updates the forwarding tables of the nodes on the shortest path. Since the considered networks are mobile, the shortest path is dynamic. Therefore the server needs to actively maintain the forwarding tables.

A routing example is shown in Figure 3.2. A connection between nodes 7 and 4 needs to be set-up. The server computes the shortest path (SP), in Figure 3.2 made by the nodes 7 - 2 - 3 - 4. The server sends forwarding table (FWT) to each node on the path. The FWTs are composed by the destination node ID of the end-to-end communication and the node ID of the next hop direct neighbour. Therefore, the nodes are now able to correctly route packets between nodes 7 and 4.

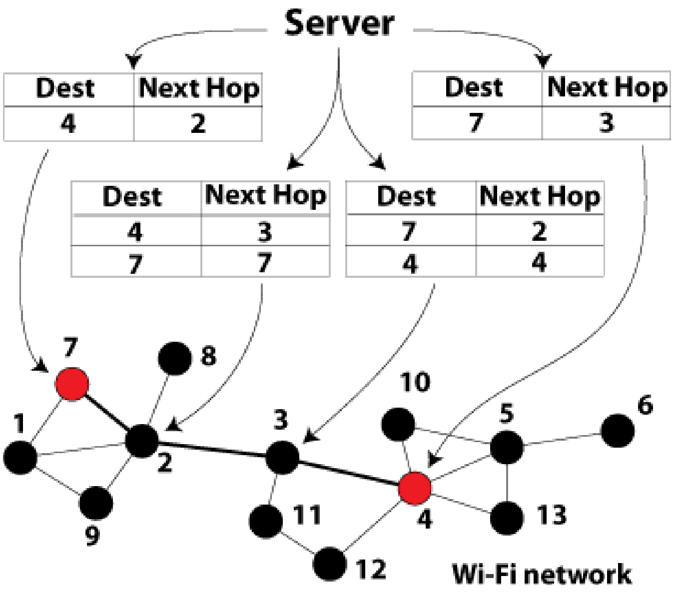


Figure 3.2: Example of routing between nodes 7 and 4. The central server computes the shortest path and sends the forwarding tables to all the nodes belonging to the path.

Even if all the nodes use the 3G link continuously, they exchange only few bytes per control packet compared to downloading several MB of data (e.g., a video stream). However, the biggest concern with the naive solution is the energy consumption of the 3G link. All the devices actively use the 3G link so the global energy consumption is high.

3.3 CDS Approach

MASS-ANET is designed using a more advanced technique, called the CDS approach, which reduces the devices' power consumption and the general 3G usage. The goal is to limit the number of devices that need to be continuously connected to the server through 3G link, as shown in Figure 3.3. The CDS approach is based on a Connected Dominating Set (CDS) algorithm applied to the Wi-Fi network. The CDS algorithm used is a variation of the one created by Alzoubi et al. [29] and described at the end of Section 2.3.

In order to correctly handle mobility in the Wi-Fi network a fast algorithm is needed. The chosen algorithm, with few enhancements, meets this requirement. It is distributed and operates on local scale considering only a maximum of 3-hop neighbours. Furthermore, the original technique of Alzoubi et al. is based on the Maximal Independent Set (MIS). A MIS is the perfect candidate for the sub-set of nodes that should exchange messages with the server over the 3G link because it is easy and fast to maintain. The MIS is a sub-set of the CDS. Therefore, two overlay networks are considered. The CDS overlay network, at the Wi-Fi level, and the MIS overlay network, at the server level. The MIS overlay network is formed by MIS nodes' connectivity within 2- and 3-hop of distance.

The main difference between the approach of Alzoubi and the one adopted in this thesis is the dominators' election technique (i.e., the set-up and maintenance phase of the MIS). The new algorithm applies the "first come, first served" rule for dominators' election. The node state is computed based on the received beacons. The nodes do not compute their state at the same time. Therefore, our technique, in the general case, is much faster than the original algorithm. However, the ID comparison method is still used to solve contentions so in the worst-case the convergence time of MASS-ANET is the same of [29]. Section 4.1.1 provides more information.

3.3.1 Basic CDS notions

Before the details of the CDS approach are explained, some basic notions and terminologies are needed.

The network is composed by three types of nodes: Dominators, Connectors and Dominatees. They are graphically represented as shown in Figure 3.4.

- Nodes belonging to the Maximal Independent Set (MIS) are called *Dominators*. They actively communicate with the central server over

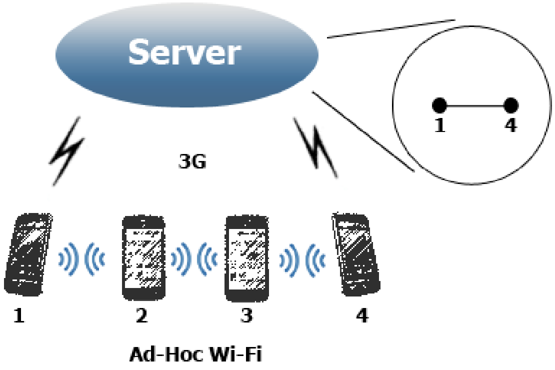
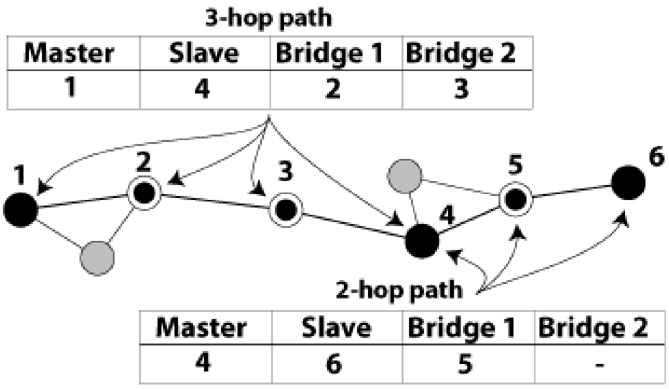


Figure 3.3: CDS approach: only some smartphones are connected through 3G to the central server.

3G to constantly update it with their dominator neighbour list. Two dominator nodes, due to the MIS properties, should not be direct neighbours.

- *Dominatee* nodes have at least one dominator neighbour.
- *Connector* nodes are dominatees which are part of at least one path to connect two dominators located at most 3-hops away.

Figure 3.4: Graphical representation of CDS nodes.



Master	Slave	Bridge 1	Bridge 2
1	4	2	3

Master	Slave	Bridge 1	Bridge 2
4	6	5	-

Figure 3.5: Example of 2- and 3-hop path.

Paths are formed between dominators through connector nodes. These paths can have 2 or 3 hops. There can be only one unique path between two dominators. Each path has a *master* dominator node ID and a *slave* dominator ID. One or two IDs of connector nodes are embedded depending on the kind of path (i.e., 2- or 3-hop path respectively). Each node keeps a list of the paths it is part of.

In a path between two dominators, the label of master and slave are selected depending on their IDs: the node having the lowest ID is chosen as master. Paths between any pair of dominators can be set-up only by the master. A slave node is the dominator with the higher ID. Figure 3.5 shows an example of a 2- and 3-hop path and the path lists in the nodes.

3.3.2 Discovery and Control

All the nodes notify the central server of their presence in the network as in the "naive" approach. Only the dominator nodes constantly keep the server updated about their 2- and 3-hop dominator neighbours. The server has a complete view of all the dominators and all the interconnections between them, as shown by Figure 3.6.

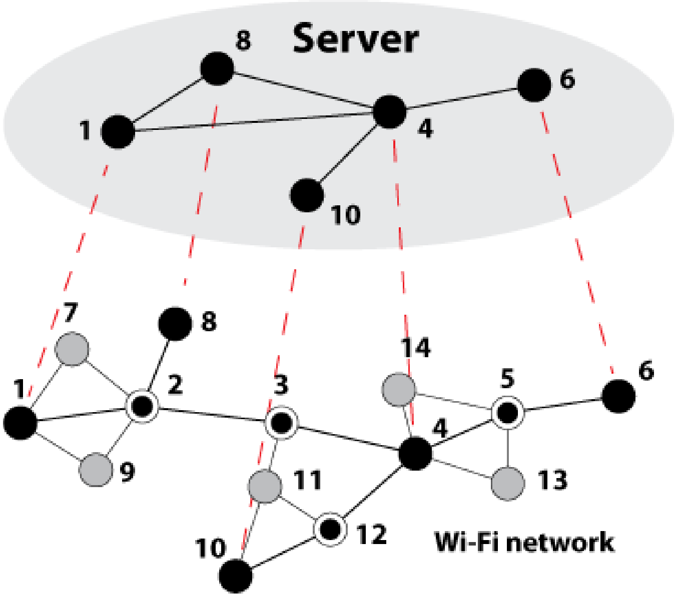
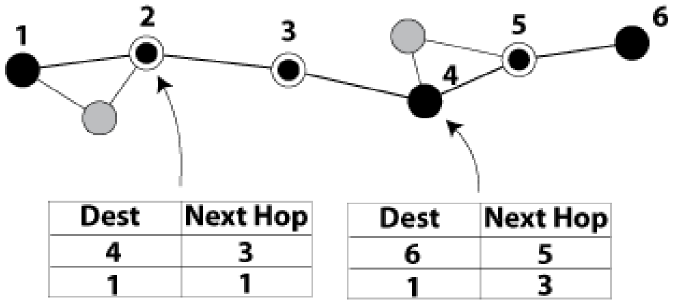


Figure 3.6: Server view of the dominators' network.

3.3.3 Routing

Compared to the naive approach, where the server has a complete view of the network, in the CDS approach the server knows and can compute the shortest path only between dominators. Therefore, the routing function is split into two parts: *intra-dominator* routing and *inter-dominator* routing.

The routing between two dominators of a 2- or 3-hop path is called intra-dominator routing. Each packet in the network contains a “destination” field with the ID of the final end-to-end destination and a “next hop” field with the ID of the next dominator on the path. A node uses the “next hop” field of the packet as destination in its path list to retrieve the ID of the next neighbour node to which it forwards the packet. Figure 3.7 gives an example of routing information, based on Figure 3.5, used by two randomly chosen nodes, 2 and 4, in order to route between dominators.



Dest	Next Hop
4	3
1	1

Dest	Next Hop
6	5
1	3

Figure 3.7: Intra-dominator routing. Example of information based on nodes’ path lists.

Inter-dominator routing is the technique used to route between several dominators in an end-to-end communication. The server computes the shortest path (SP) between the source and the destination dominators and sends the forwarding tables (FWTs) to the dominators on the path. The FWTs are composed by the end-to-end destination ID and the ID of the next dominator on the SP. A dominator involved in a transmission, changes the “next hop” field of the received packets, according to the FWT received from the server. The intra-dominator routing can now be applied until the packet reaches the next dominator.

Figure 3.8 shows an example of inter-dominator routing. The source and the destination dominator nodes are 1 and 6, in red in the figure. The server computes the shortest path, which is through the dominator 4. The FWTs are sent to the dominators on the path. The dominators are now able to insert the correct “next hop” dominator ID in the packet field and intra-dominator routing can be applied by both dominators and connectors.

The central server always exchanges control messages with both the source and the destination of a communication before the connection set-up phase. During the described process, the destination can refuse the connection with

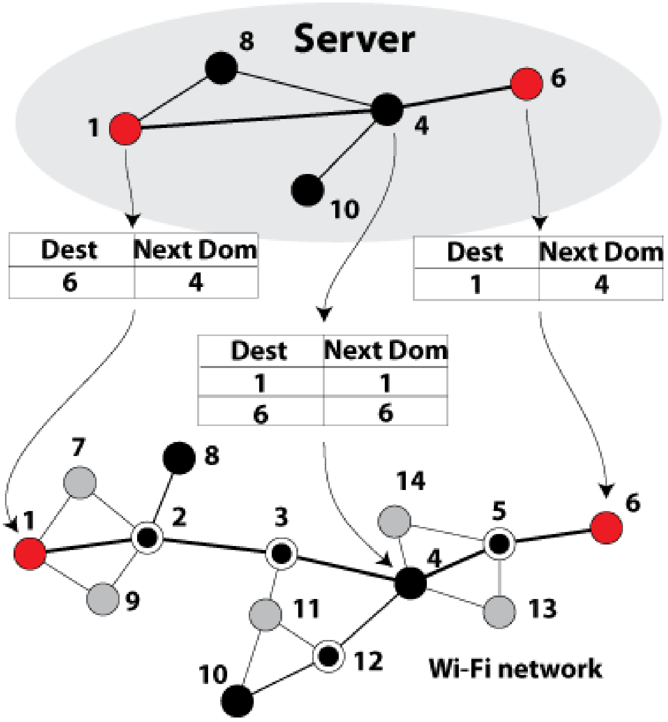


Figure 3.8: Inter-dominator routing. Example of forwarding tables sent by the server to the dominators in the shortest path between dominator nodes 1 and 6.

the source. Moreover, the server can directly decline the connection request by the source if the destination is further than a hop count threshold.

If the source and/or the destination node of a communication is not a dominator, it sets a special bit, called Presence Node (PN), in its beacon. If a dominator receives a beacon, from a neighbour node, with the PN bit set, it will notify the server about the position of the node it received the beacon from. The server is now able to compute the shortest path and check if the destination is in range of the source.

3.3.4 CDS Control Messages

The behaviour of the CDS algorithm is now explained in more details. This section describes the control messages exchanged in the Wi-Fi network needed to build and maintain the CDS.

Three kinds of control messages are exchanged: Beacons, Link set-up messages and Link tear-down messages.

]

Beacons

Beacons are used by each node to show their presence in the neighbourhood. If a node is not a dominator (i.e., it has at least one advertised neighbour dominator) it can create or forward `D_INFO` messages. `D_INFO` messages are piggybacked on beacon payload. Non-dominator nodes generate new `D_INFO` messages for each new beacon and for each neighbour dominator. A `D_INFO` message, when created, includes a dominator neighbour ID, called “origin”, the ID of the creator and a Time To Live (TTL) of 1. `D_INFO` messages can be forwarded at most once by another non-dominator node that appends its ID and changes the TTL to 0. A node that receives a `D_INFO` message can extrapolate the reverse path between itself and the “origin” dominator of the `D_INFO` message. Dominators do not broadcast any payload in the beacons but collect all the `D_INFO` messages to gain knowledge of the 2- and 3-hop dominator neighbours. If a dominator has a smaller ID with respect to another one at 2- or 3-hop of distance, it is the master of the path that it can set-up between them.

Path set-up messages

Path set-up messages are sent by dominators that are masters of a possible path. A dominator computes all the paths, choosing the optimal one, among its slave dominator neighbours at 2- or 3-hop distance using the collected `D_INFO` messages from the neighbour nodes.

1. When a dominator sets up a path with another dominator, it looks inside the collected `D_INFO` messages for the reverse nodes’ path to reach that dominator.
2. Then, it sends the set-up message with the path information (master, slave and connector IDs) to the first node of the reverse nodes’ path.
3. The first node is now notified to be part of a path so it becomes a connector and records the path in its path lists. Moreover, it forwards the set-up message to the next connector, if it is a 3-hop path, otherwise to the slave dominator.
4. The second connector, in case of a 3-hop path, follows the same rules as the first one.
5. The slave dominator is now notified about the new path with the master one.

Path tear down messages

Path tear down messages are sent similarly as the set-up ones. They are used to remove a path from the nodes on it. The message is sent by the master dominator only if it discovers a shorter path, from 3- to 2-hop, to the slave. In a mobile network, paths can often become disconnected because nodes are not in range any more. If a node is an edge of a path and that path disconnects, it will just remove the respective entry from its path list. A path disconnection is detected by a threshold on the statistics of beacons received. Details are provided in Section 5.2.2. A connector node has two links belonging to a path. If one of these links disconnects, the connector will notify the disconnection with a tear down message to the side of the path that is still connected. The message is forwarded by the second connector if any. Therefore, in case of disconnection, this message is created only by one of the connectors and if necessary, forwarded by the second one.

Working Example

In order to better understand the CDS management, an example network is proposed with the related graphic representation in Figure 3.9 and in Figure 3.10. At the bootstrap phase, seven nodes are in place with the state set to dominatee and no knowledge about the network. Nodes update their state and `D_INFO` messages, and then the beacon is sent. The algorithm's computation and beaconing order is randomly permuted in every round. In this example we assume that no parallel algorithm computation is executed. The first beaconing round order could be: 3-7-6-1-2-5-4.

1. Node 3, initially dominatee, knows that it does not have any dominator in the neighbourhood. Therefore, it becomes a dominator and sends its new state, which is received by node 2.
2. Node 7 does the same as node 3.
3. Node 6, which has just received the beacon from dominator 7, sends its dominatee state plus the `D_INFO` about 7.
4. Node 1, as initially 7 and 3, does not know anything so it becomes dominator and broadcasts its beacon.
5. Node 2 received the beacons from dominators 1 and 3, so broadcasts its dominatee state and the `D_INFO` regarding 1 and 3 to nodes 1, 3 and 4.
6. Node 1 at this point infers a path between itself and node 3, a dominator with higher ID, so it is the master and can create the link through node 2. Node 2 becomes a connector and will disseminate its new state at the next round.
7. Node 5 knows the dominatee node 6 and the node 7 through it. It is in the condition for being a dominator and also for being the master of

the path between itself and 7. Therefore, it will create the path with 7 using 6 as connector.

8. Finally 4 is the last node that broadcasts its dominatee state, the `D_INFO` of 5, of 7 and forwards the `D_INFO` of node 2 regarding the dominators 1 and 3.

9. Both 5 and 7 receive that `D_INFO` message but as their ID is higher than 1 and 3 they cannot create any link because they are slave of that.

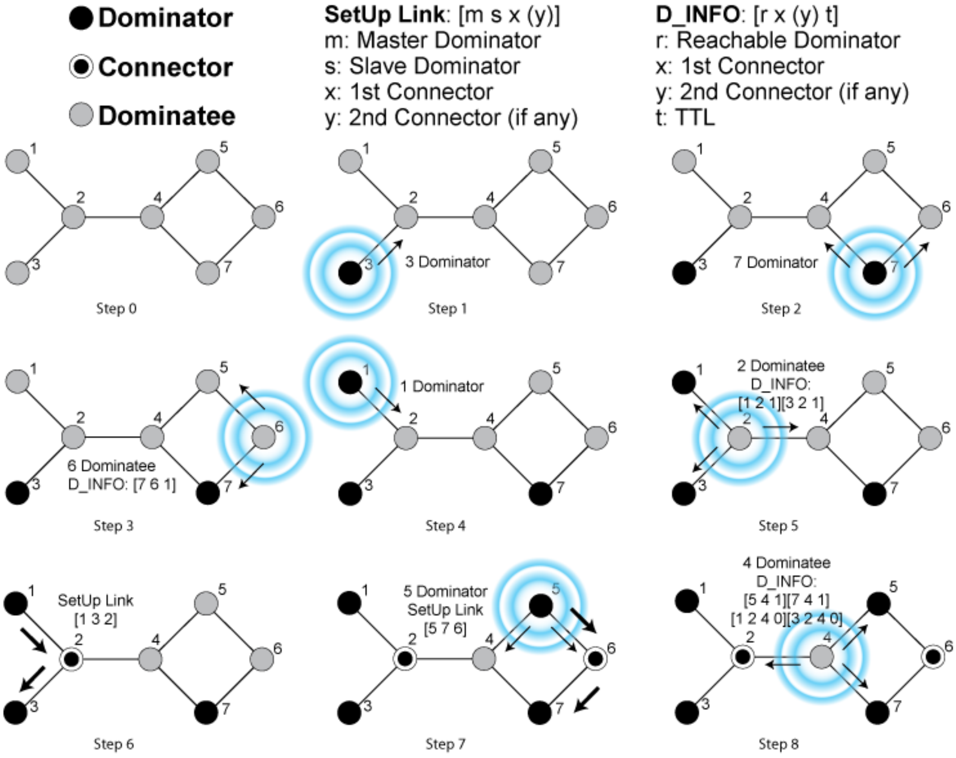


Figure 3.9: CDS algorithm working example, first beaconing round.

In the second iteration, regardless of the beaconing order, nodes 1 and 3 receive the `D_INFO` messages from 2 about nodes 5 and 7 and they, both masters, create the respective four links with 5 and 7 through the connector nodes 2 and 4. The final configuration is shown in Figure 3.10.

Even if the optimal number for both dominators and connectors is two, due to the random computation and beaconing order, the resultant number is higher. In this example the algorithm converges with two iterations. Potentially, a few more iterations are required to optimize the number of

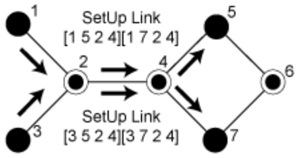


Figure 3.10: CDS algorithm working example, second beaconing round.

connectors by removing the redundant ones as the number 6, but this is outside the scope of this research. Anyway, in mobile networks where links are often broken, having more connectors is useful to maintain network connectivity.

3.3.5 Operations

Set-Up a connection

Two kinds of device communications are possible between a source A and a destination B:

1. Device - Device
2. Device - Access Point (AP)

The destination device B can be either the smartphone of a specific owner or the nearest free internet access point (AP). In order to start a communication the source node A sends to the central server the connection request to an AP or to another device. Moreover, if A is not a dominator it sends also its position.

An access point is set to always be a dominator. Therefore, the central server can immediately compute the distance, in dominator hop count, between the AP and the requesting device. Based on the APs' availability, distance and paths congestion, the server evaluates the feasibility of the connection and immediately replies to A if it is not possible.

If the connection is feasible, the server forwards the connection request from the source A to the destination B. B can either accept or refuse the connection. The request answer of B is forwarded by the server to the source A. Figure 3.11 shows the connection operations between a source device A and an AP so far described.

If the connection is refused the communications are concluded. If accepted, and the destination B or the source A are not dominators, they need to be visible to the central server in order to compute the shortest path. They can be turned into dominators or, in the adopted solution, a special rule explained in Section 3.3.3 is applied setting a special bit in the destination's beacons.

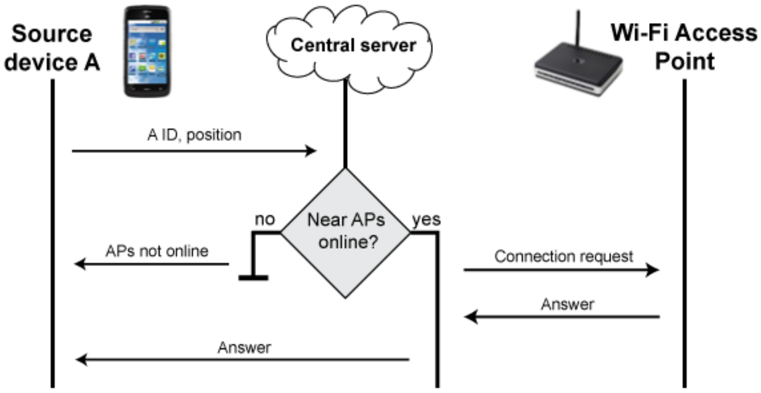


Figure 3.11: Device-AP connection set-up, first phase.

After the destination has accepted the connection, the central server, as explained in Section 3.3.3, computes the shortest path, through dominators, between the source and the destination and it sends the correct Forwarding Table (FWT) to the nodes on the shortest path.

Moreover, a 2-way acknowledge message is sent through the Wi-Fi to be sure that both the inter- and the intra-dominator routing are working properly. The first acknowledge starts from the destination back to the source, which then sends a reply to the destination. The end-to-end connection is now ready to be used. The last phase of the connection set-up is shown in Figure 3.12.

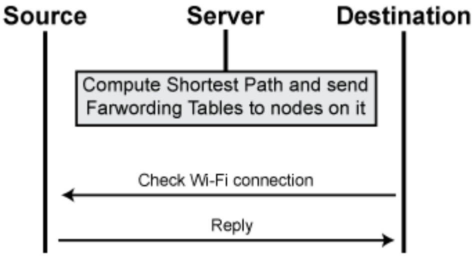


Figure 3.12: Connection set-up, last phase.

A smartphone-smartphone connection, as showed in Figure 3.13, requires some extra efforts. If the destination is a specific device (e.g., smartphone), the source A sends to the central server, together with the connection request, also the ID of the destination B. The central server, before proceeding with the connection set-up, needs to check if B is online and it needs to retrieve B's position if B is not a dominator.

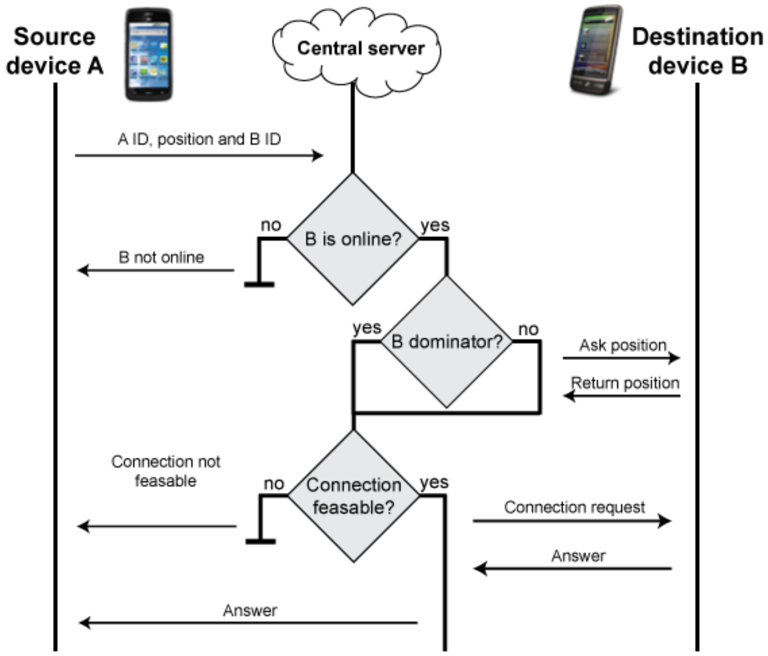


Figure 3.13: Device-device connection set-up, first phase.

Maintaining a connection

In the maintaining phase the central server, which constantly receives updates from the dominator devices, continuously computes the shortest path of the online connections. It uploads the new forwarding tables to the new devices in the path while removing the old ones. In case the network between the source and destination nodes disconnects, the central server notifies the nodes and the connection can still use the 3G link as a backup.

Tear Down a connection

The source or the destination can notify the server that the communication is finished. The server notifies the other edge and it removes the respective FWT entries from all the dominator nodes involved in that connection.

Chapter 4

ANALYSIS

In this chapter the MASS-ANET approach is analysed. In the first section we introduce the metrics studied in this chapter and the simulator used to obtain initial results. Section 4.2 studies the influence of the Maximal Independent Set (MIS) size on the power consumption. Sections 4.3 and 4.4 investigate the message complexity of the Wi-Fi and 3G channel, respectively. An example scenario of offloaded 3G data and energy consumption is presented in Section 4.5 using the result from Sections 4.2 - 4.4.

4.1 Introduction

In order to evaluate the behaviour of MASS-ANET we investigate two key metrics:

1. The performance of MASS-ANET in terms of time complexity.
2. The energy cost associated with MASS-ANET.

We consider several application scenarios with the support of a discrete-time event simulator.

4.1.1 Time complexity

The convergence time of MASS-ANET is related to the behaviour of the algorithm in mobile scenarios. Algorithms with a short convergence time are more resilient to network dynamics.

The time complexity of Alzoubi et al.'s algorithm [29] is $O(n)$ due to the Maximal Independent Set (MIS) construction technique. MASS-ANET also achieves $O(n)$ in the worst-case scenario due to the parallelism of the algorithm. However, the optimization of the MIS construction technique leads to a much faster average convergence time.

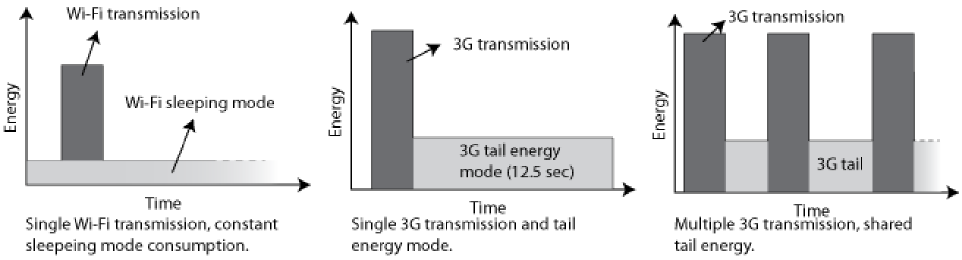
The original algorithm [29] uses an initial state for the nodes called "candidate". In this state each node just collects information about the neighbour

IDs. In order to change state, the nodes with higher IDs need to wait until all the lower ones change state. In the worst-case scenario all nodes are arranged in either ascending or descending order along a line (i.e. max node degree Δ of two).

In our adaptation, a “first come, first serve” rule is applied. No candidate state is considered. The lower ID rule is only used for contention between two dominators. Using the previous worst-case example, starting with all the nodes in dominatee state, all the nodes, one by one, will try to become dominators and broadcast their new state. In the worst-case the nodes compute their state synchronously because the events are parallel. Therefore, all the nodes become dominators and the contention rule is applied. In a line topology with IDs in ascending or descending order all but the lowest ID node change back to the dominatee state. Then, the process is repeated determining one node’s state per step. Therefore, the worst-case convergence time is $O(n)$. However, usually the nodes do not compute their state exactly at the same time but they are desynchronized. Therefore, the convergence time of the general case is much faster because the quickest nodes become immediately dominators.

4.1.2 Energy consumption

Energy consumption is an important metric in mobile environments. Both 3G and Wi-Fi radios consume energy in order to keep MASS-ANET alive. The energy is related to the number of messages exchanged and their frequency. Therefore, in the next sections we will investigate the message overhead in order to study the energy consumption of both radios.



Single Wi-Fi transmission, constant sleepeing mode consumption.

Single 3G transmission and tail energy mode.

Multiple 3G transmission, shared tail energy.

Figure 4.1: Wi-Fi and 3G radio energy consumption outline.

The 3G radio is also affected by the tail energy. In order to reduce the influence of the tail energy we limit the number of nodes that use the 3G link. The nodes allowed to use 3G are only the dominators (i.e., MIS). Therefore, even if the number of 3G messages between the CDS enhancement and the naive approach are the same in each beaconing period, the tail energy in the dominators is shared among more messages. Non-dominator nodes do not use 3G for network maintenance. Moreover, the considered beaconing period of 1 second is less than the 3G radio backoff time (i.e., about 12.5s) so the tail energy is shared even in low-mobility among 3G messages. Figure 4.1 shows a schematic representation of the energy consumption of a single Wi-Fi and 3G transmission and multiple 3G transmissions with shared tail energy.

4.1.3 Simulator

In order to verify the MASS-ANET approach we used the discrete-time event simulator NetLogo. NetLogo is a "multi-agent programmable modelling environment for simulating natural and social phenomena" [48]. Both the naive approach and MASS-ANET, as described in the design Chapter 3, are fully implemented in the simulator with the exception of end-to-end connections among nodes.

In discrete-time event simulators, the simulation time is divided into rounds. Each time round represents a beaconing period. A beaconing period, in our network, lasts 1 second. The nodes are randomly deployed in the simulation area. At each round, in a random order, the nodes execute the steps of the algorithm. They update their status based on the beacons received from the neighbours until that moment, and they broadcast a beacon with the updated status.

All the simulations use a planar square surface area of 1km^2 and a perfect circular radio range with a radius of 100m. The links are assumed to be completely reliable. In every graph, each simulation point represents the average of 25 executions.

In order to simulate mobile networks, we used the Random WayPoint model (RWP) introduced in [10]. However, we are aware of the known problem with the RWP model about the non-uniform node density reported by [7, 28], caused for example by the border effect among others. In order to immediately reach the desired node average speed, constrained between $V-min$ and $V-max$, we removed the waiting time between two consecutive waypoints from the model. The algorithm used is:

```
if currentPosition != destination {
   moveTo(destination,speed);
}else{
   destination = randomDestination;
   speed = randomSpeed(Vmin, Vmax, granularity);
}
```

Simulations are used to obtain several data for our analysis including:

- The number of Wi-Fi/3G messages exchanged.

- The average number of paths in the network: 1-hop links for the naive approach and 2- and 3-hop paths for MASS-ANET.

- The lifetimes of links, 2- and 3-hop paths and 2- and 3-hop dominator connectivity.

4.2 Maximal Independent Set (MIS)

We chose the CDS algorithm of Alzoubi et al. [29], as starting point of MASS-ANET, due to the usage of the Maximal Independent Set (MIS). A feature of the MIS, within a defined area and determined radio range, is that it will be constant in size regardless the network density. Only the dominator nodes, which are the MIS, use the 3G link to maintain MASS-ANET.

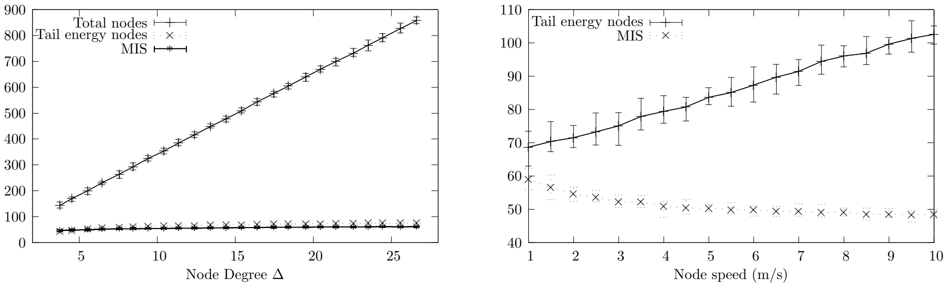
Legend	
n_d	`Total number of dominators.`
A	`Surface area (m`^{2}`).`
r	`Wi-Fi link range radius (m).`

$$n_d = \frac{2A}{\pi r^2} \tag{4.1}$$

An upper bound to the maximum number of dominators n_d in a planar surface of area A with nodes radio range radius r is given by the Equation 4.1. The surface area is divided by the circular area of the half of the radio range. In fact, two nodes are independent if their distance is at least $r/2$.

In Figure 4.2 (a) the total number of nodes and the number of dominators are shown with respect to distinct node degree Δ values and an average speed of 1.2m/s. The total number of nodes in the simulation area is linearly related to Δ. The number of dominators, instead, is nearly constant. Therefore, the network density does not have any influence on the total number of nodes that use the 3G connections. The result, n_d, of Equation 4.1 using the simulation's parameters, is 63.6 dominators and confirms the result values from the simulations shown in Figure 4.2 (a).

We study the effect of mobility on tail energy. A node is in tail energy mode for 12.5 seconds after a 3G transmission. Only the MIS nodes are allowed to use the 3G radio. However, due to the mobility of the network the nodes that leave the MIS are affected by the tail energy up to 12.5 seconds afterwards. Therefore, the nodes in tail energy mode include the MIS and the nodes that just left the MIS. The number of nodes in tail energy mode, but not in the MIS, increases with higher density and/or mobility.



(a) (b)

Figure 4.2: Total number of nodes, of dominators and of nodes with 3G radio in tail energy mode (a) with variable node degree Δ and node speed of 1.2m/s (b) with variable node speed and $\Delta \simeq 15$.

In fact, Figure 4.2 (b) shows the same metrics of the graph (a) with a fixed $\Delta \simeq 15$ and variable speed. As expected, increasing node speed produces an increasing difference between MIS size and total nodes in tail energy mode.

The number of nodes in tail energy mode is only a small fraction of the total number of nodes, even with a speed of 10m/s. Therefore, the energy consumption due to 3G tail energy of nodes not involved in any 3G communications is still negligible.

4.3 Wi-Fi overhead

There are two kinds of Wi-Fi messages: Beacons and Path Control (PC) messages. Estimating the number of beacons broadcasted is trivial because each node beacons periodically. In this section we investigate the network overhead only in terms of PC messages exchanged by CDS nodes or link set-up and tear down messages.

Legend	
Pt_k	`Average lifetime for the k-hop paths (s). It depends on network parameters such as number of nodes, surface area, radio range radius and mobility pattern.`
Pn_k	`Average number of k-hop paths in the network. It depends on number of nodes, surface area, radio range radius and mobility pattern.`
B_k	`Average number of broken links in a k-hop path in the network. It depends on number of nodes, surface area, radio range radius and mobility pattern.`
W_m	`Average number of Wi-Fi messages to maintain the network per beaconing period (1 second).`

$$W_m = \frac{4Pn_2}{Pt_2} + \frac{6Pn_3}{Pt_3} - B_2 - B_3 \tag{4.2}$$

Given the average changing rate of k-hop paths in the network, Equation 4.2 shows the number of Wi-Fi messages exchanged to maintain the network. 2- and 3-hop paths are multiplied by 4 and 6 respectively, which are the total number of messages needed to set-up and to tear down a single k-path. Sometimes a path is torn down because a link is broken. Therefore, the Wi-Fi messages are sent only on the connected side of the path. B_k subtracts the average unsent messages due to link disconnections from the final results.

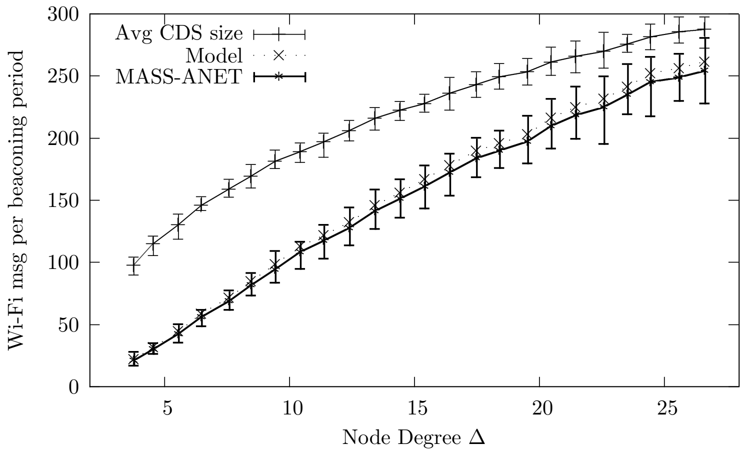


Figure 4.3: Wi-Fi messages per beaconing period and CDS size.

Three functions are shown in the Wi-Fi overhead per beaconing period in Figure 4.3 with respect to different Δ. The first one represents the size of the CDS. The CDS size is composed of dominator and connector nodes. Only the nodes belonging to the CDS use the Wi-Fi channel to exchange PC messages. The second value depicts Equation 4.2. The last function is the number of Wi-Fi messages resulting from the simulations. In our scenario each CDS node sends on average less than one Wi-Fi PC message per beaconing period.

In Figure 4.4 the Wi-Fi messages and CDS size are plotted versus node speeds with a constant Δ of 15. At about 2m/s all the CDS nodes send one PC message per beaconing period. Above 2m/s the PC messages are more than one per node. The decrement of the CDS size with the increment of the speed is due to the border effect of the Random WayPoint Mobility model. In fact, most of the nodes are concentrated in the center leaving the area near the borders empty.

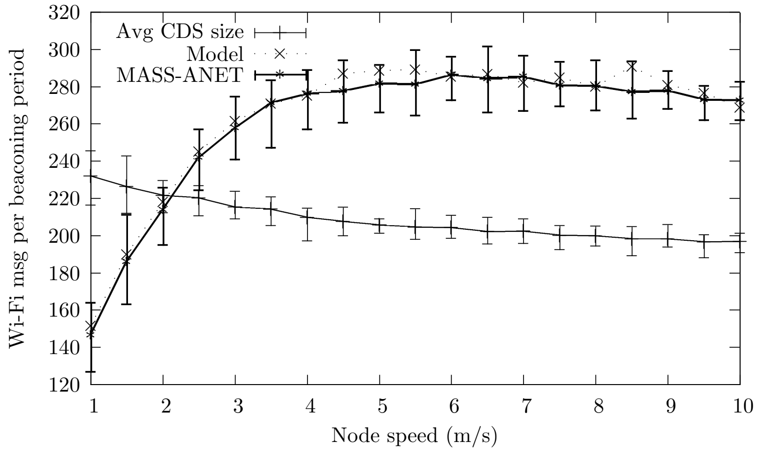


Figure 4.4: Wi-Fi messages per beaconing period and CDS size.

4.4 3G overhead

The 3G message overhead is composed of the messages due to the maintenance of the mobile network and the messages to support end-to-end routes.

4.4.1 Network maintenance

Dominator nodes update any change to their 2- and 3-hop dominator neighbour list to the central server over 3G. Therefore, the server maintains a view of the overlay network made by the dominators (i.e. MIS). The connectors between two dominators could change over a beaconing period. However, a change in path connectors, between the same two dominators, does not affect the MIS connectivity at the server. Therefore, no 3G messages are needed. Moreover, because a path between two dominators can change without affecting the connectivity, the dominator's connectivity lifetime Dt_k is higher than the k-hop path lifetime Pt_k considered in the Wi-Fi overhead.

Legend	
Dt_k	`Average lifetime for the k-hop dominator's connectivity (s). It depends on the number of nodes, surface area, radio range radius and mobility pattern.`
G_m	`Average number of 3G messages needed to maintain the network per beaconing period (1 second).`

$$G_m = \left(\frac{Pn_2}{Dt_2} + \frac{Pn_3}{Dt_3} \right) \cdot 2 \cdot 2 \tag{4.3}$$

The average number of 3G messages needed to maintain the network for each beaconing period is given by Equation 4.3. The average updates exchanged with the server depends on the total number of paths in the CDS, considered as dominator's connectivity, and their lifetime. The 2- and 3-hop changing paths are multiplied twice by two. In fact, both sides of a path send an update to the server. Moreover, for each path change, two updates are necessary to remove the old path and to add the new one.

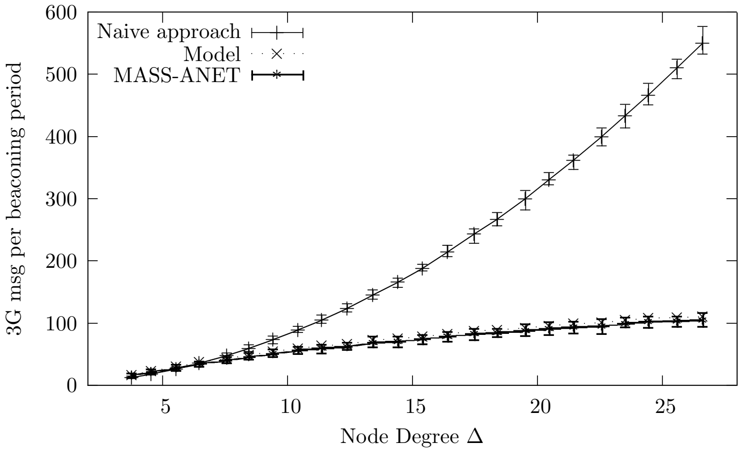


Figure 4.5: 3G messages per beaconing period.

Figure 4.5 shows the 3G maintenance message overhead with respect to an average Δ. The overhead is reported as 3G messages per beaconing period. The result of Equation 4.3 almost overlap the number of messages simulating MASS-ANET.

The graph also shows the 3G messages used, in the same networks, by the naive approach where all the nodes are connected to the central server over 3G. With low network density the messages due to the naive approach are almost equal to MASS-ANET. However, the messages of the naive approach increase rapidly with the increment of the average node degree Δ overtaking MASS-ANET.

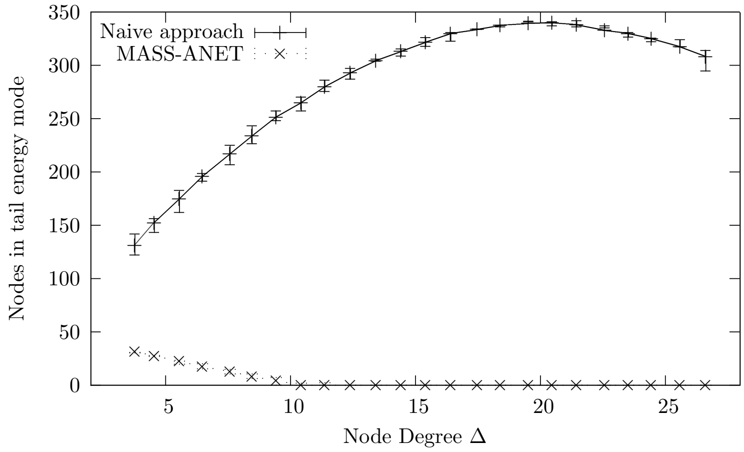


Figure 4.6: Number of nodes in 3G tail energy mode per beaconing period.

The average number of messages per node are used to estimate the network energy consumption due to the 3G tail energy for network maintenance.

Figure 4.6 shows the number of nodes affected by 3G tail energy per beaconing period. In the naive approach the nodes influenced by the tail energy increase with the increment of the network density. However, the nodes in tail energy mode start decreasing as soon as the number of 3G messages increases faster than the number of nodes as shown in Figure 4.5 and 4.2(a). MASS-ANET instead concentrates all the messages in a small number of nodes. In fact, after a Δ of 10 one or more 3G messages per beaconing period are sent by the MIS. Therefore, the 3G channel is used efficiently and no tail energy is consumed.

Figures 4.7 and 4.8 show the same metrics of the two previous figures (4.5 and 4.6) using a network of about 500 nodes (Δ=15) with respect to a variable node speed. The 3G messages needed by the naive approach are severely affected by the node speed while the difference is not so evident in the MASS-ANET. The nodes belonging to the MIS in MASS-ANET are never affected by the tail energy. In the naive approach, the tail energy influences nodes only before 3m/s. In fact, after 3m/s the 3G messages required by the naive approach per beaconing period are higher than the number of nodes in the network (500). Therefore, each node sends at least one 3G message per beaconing period, forcing the radio to be on at all times.

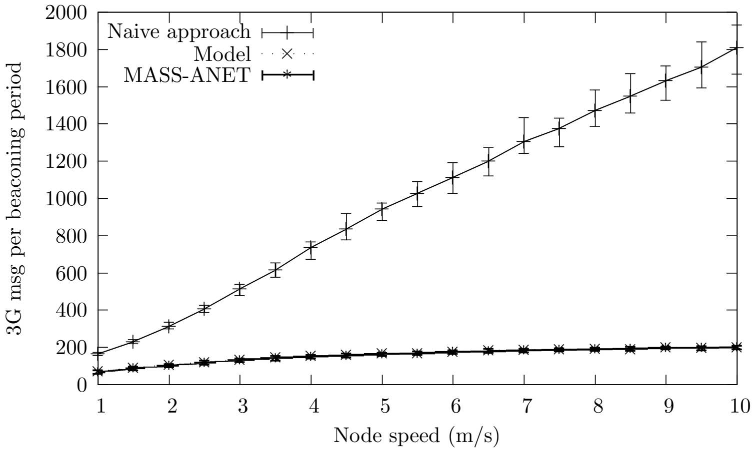


Figure 4.7: 3G messages per beaconing period.

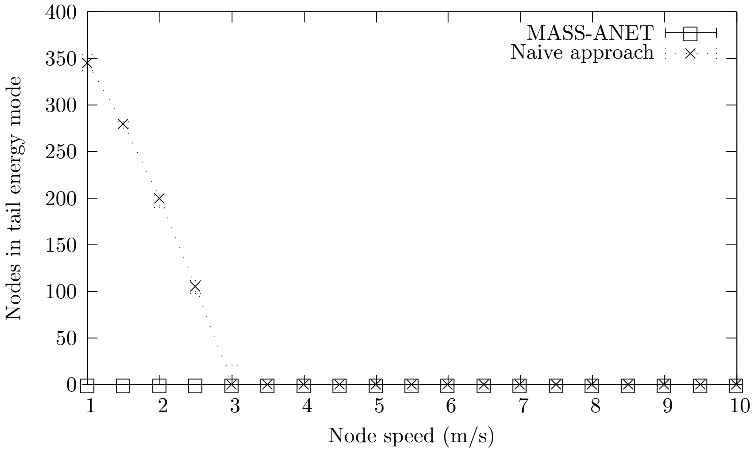


Figure 4.8: Number of nodes in 3G tail energy mode per beaconing period.

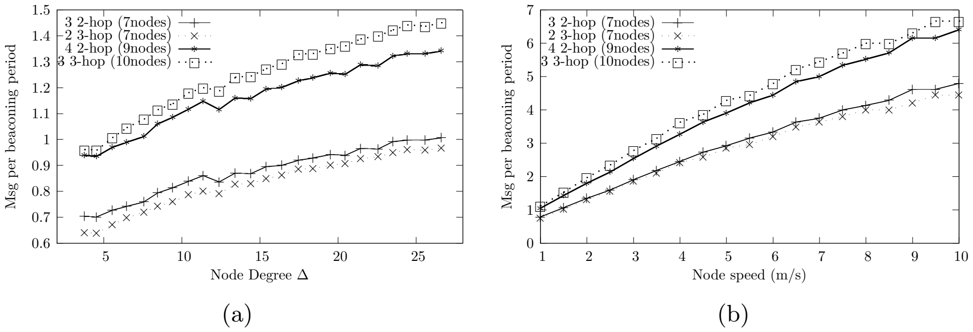
4.4.2 Routing maintenance

Two routing levels, as explained in Section 3.3.3, are needed in end-to-end connections. Intra-dominator routing handles messages exchanged between two dominators connected through a 2- or 3-hop path. No additional information is required for intra-dominator routing besides the Wi-Fi Path Control (PC) messages exchanged to maintain the CDS. Instead, inter-dominator routing is used to globally forward messages inside the network, among dominators. Therefore, extra 3G messages from the server are required to globally control end-to-end connections.

Legend	
n	Total number of nodes.
m_p	Number of nodes in a path p.
Δt	Total time (s).
p	Some path $\{P_0, P_1, P_2, ..., P_x\}$, $P_x \in n$.
$L_{p,k}$	Number of k-hop paths in p.
$G_{p,k}$	Total Number of 3G messages to maintain an end-to-end connection with m_p nodes and k-hop paths.

$$G_{p,k} = L_{p,k} \cdot 4 \cdot \frac{\Delta t}{Dt_k} \tag{4.4}$$

Equation 4.4 represents the number of messages sent over 3G to maintain an end-to-end connection during Δt seconds. $L_{p,k}$ is the number of k-hop paths in the end-to-end connection. It is multiplied by 4 because two messages are sent to insert the new FWTs at the edges of each path and two more are sent to remove the old FWTs.



(a)

(b)

Figure 4.9: 3G messages needed to maintain routing path of several 2- and 3-hop with respect of node density Δ (a) and node speed (b).

Figures 4.9 (a) and (b) show the number of messages needed to maintain end-to-end routing paths of varying lengths according to Equation 4.4. The number of messages needed per beaconing period increases with node speed and network density. Moreover, irregardless of the kind of paths (2- or 3-hop paths) used in an end-to-end connection the number of messages required are nearly the same. Therefore, no additional rules are required in the MASS-ANET routing algorithm.

4.5 Example

The purpose of this thesis is to offload 3G data traffic to Wi-Fi. In order to determine the amount of traffic effectively offloaded, a simple example scenario is presented. A download of 5.85MB is compared between normal 3G usage and MASS-ANET. We assume a full mobile MASS-ANET composed of 385 nodes (average $\Delta = 11$ and average number of dominators = 55) with an average node speed of 1.2m/s, area of 1km^2 and Wi-Fi radio range of 100m. The available Wi-Fi bandwidth for bulk data transfer is assumed to be 400Kbps. We assume also a chain of 7 nodes (6 hops) where the outer nodes are a requesting smartphone and an Access Point.

The data presented is retrieved from the simulations. Moreover, the sizes of the packet payloads used in the next calculations are an overestimation of the ones used in the prototypes. Two bytes are used to represent a node ID.

In the normal case all the 5.85MB of data are transferred over 3G. In MASS-ANET the data is completely transferred over Wi-Fi in about 120 seconds[1]. However, the 3G network is still used to keep MASS-ANET alive. In 120 seconds MASS-ANET exchanges 7,113 3G maintenance messages of a maximum 33 bytes. Therefore, the 3G maintenance overhead for the 120 seconds is 229.2KB.

We now consider the 3G overhead due to the routing maintenance. Two possible scenarios are considered as showed by Figure 4.10:

- Four dominators with three 2-hop paths.
- Three dominators with two 3-hop paths.

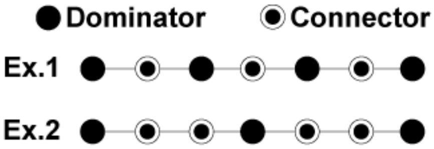


Figure 4.10: Two example scenarios with a chain of 7 nodes.

---

[1] The 5.85MB are transferred over Wi-Fi with a bit-rate of 400Kbps

The lifetime for 2-hop dominator connectivity obtained by simulation is 14s and for the 3-hop is 10s. The size of a packet to add a forwarding table (FWT) is 7 bytes, and 5 bytes to remove it. The 3G overhead in terms of size can be computed using Equation 4.4 and multiplying by the size of the packets. In the first scenario it is needed to add and remove the FWTs, on the 4 dominators, 8.5 times per path for a total of 612 bytes, while in the second scenario the FWTs are added and removed 12 times on 3 dominators for a total of 576 bytes. The 3G routing maintenance for just one route is negligible compared to the entire 3G network maintenance. Therefore, it is not included in the final 3G overhead.

In conclusion MASS-ANET transfers on 3G 229.2KB versus the initial 5.85MB. Thus, MASS-ANET saves 96.2% of 3G traffic. More downloads increase the efficiency because the 3G overhead is almost constant.

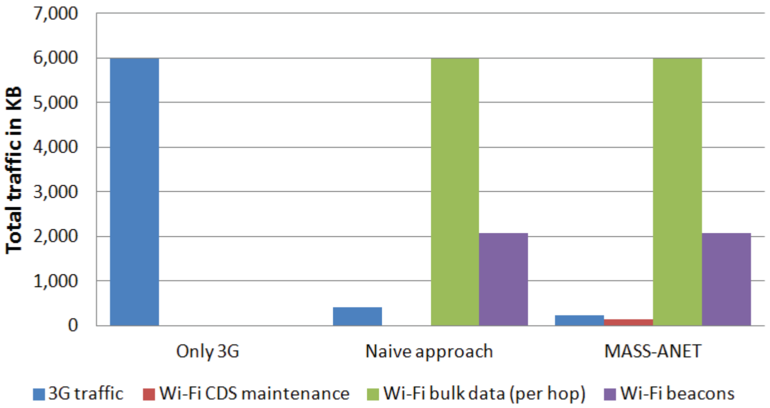


Figure 4.11: Network overhead in traffic data size. The values of the Wi-Fi bulk data need to be multiplied by the 6 hops of the end-to-end connection.

The total network overhead is the sum of both 3G and Wi-Fi. 14,111 Wi-Fi messages are exchanged to maintain the network during 120 seconds excluding the Wi-Fi beacons. With 10 bytes per message, a total of 137.8KB of data is needed for CDS maintenance. 2,073KB is the total beacon overhead for all the 385 nodes using 46 bytes per beacon.

The constant complete 3G + Wi-Fi overhead to maintain the network over 120 seconds is 2,440KB, that is the 40.7% of the original 5.85MB. With two transfers of 5.85MB the overhead decreases to 18.5%. With three download, the offloaded traffic overcomes the traffic required for MASS-ANET maintenance. In fact, each new end-to-end routing path in the network increase the overall overhead just by about 1KB over 3G. Therefore, the efficiency will rapidly increase with more connections to APs.

A message overhead comparison is shown in Figure 4.11. The columns of the Wi-Fi bulk data need to be multiplied by the 6 hops of the end-to-end connection. The difference in 3G traffic between naive and MASS-ANET approach is not so high because the number of 3G messages sent with a $\Delta = 11$, as shown by Figure 4.5, is relatively close. However, the 3G messages of the naive approach increase much faster with respect to MASS-ANET. Therefore, with higher network density the 3G traffic difference is clearly higher.

Even if a huge amount of traffic is offloaded from the 3G, the energy consumption still needs to be discussed. Using the energy model of [39] a normal 3G download of 5.85MB consume 161 Joules of energy (153.2 Joules for the transfer plus 7.75 Joules for the tail energy).

In order to compute the 3G and Wi-Fi energy consumption using MASS-AENT we used data from [39]. However, the smallest message size considered in [39] is 1KB while MASS-ANET only uses a few bytes per message. Therefore, routing messages are not taken into account because they are already considered inside the 1KB per messages due to the network maintenance.

According to [39] one message of 1KB per second per nodes consumes 3 Joules. The total 3G consumption on all the 55 dominator nodes over the transfer time of 120 seconds gives a total of 19,800 Joules. Only an average of 10 nodes per beaconing period are affected by tail energy due to the MIS alteration for a total of 6.2 Joules. Since the Wi-Fi energy consumption is not affected by tail energy, we use the total size of messages sent, including beacons, in order to apply the energy model of [39]. The total size is strongly dependent on the size of the packet. Assuming a routing path of 7 nodes, the Wi-Fi consumption due to beacons and CDS messages is 21.3 Joules, 2,268 Joules are needed to keep the Wi-Fi radio on over all the nodes per 120 seconds and finally 293 Joules for the bulk data transmission in the routing path (42 Joules per node).

The complete network energy consumption, including end-to-end connections, is 22,382 Joules or 58 Joules, on average, per node.

We mostly considered the 3G technology in our analyses because it is nowadays the standard *de facto* on mobile communications due to the higher bandwidth compared to 2G technology. However, the 3G energy consumption is very high and MASS-ANET does not require high bandwidth to transfer a few bytes of control traffic. Therefore, to further reduce the energy consumption we can use 2G GSM technology as the global channel because its energy consumption is less than half of the 3G.

Using GSM technology, the consumption due to the server communications is reduced to 6,602 Joules. Adding the Wi-Fi messages, the total consumption is 9,372 Joules with an average of 24 Joules per node. Figure 4.12

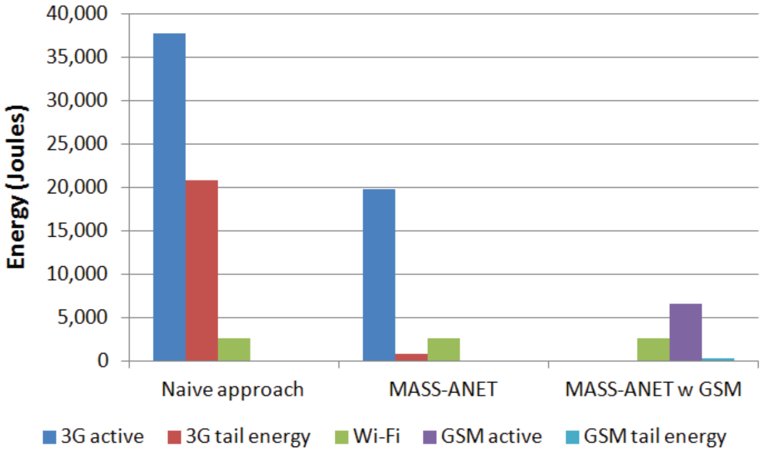


Figure 4.12: Network energy consumption.

shows a comparative between the naive approach and MASS-ANET with respect to the energy consumption.

Summing up, the 3G traffic is highly offloaded to the Wi-Fi and in the example scenario the average MASS-ANET energy consumption per node is 1/3 (3G) or 1/7 (GSM) with respect to the consumption of a single node 3G download. Each routing path hop adds 42 Joules to the total Wi-Fi energy consumption while the normal 3G radio consumes 161 Joules. Therefore, when a high number of nodes are downloading at the same time, the total energy consumption of MASS-ANET can be equal or less than the normal usage of 3G only if the routing paths are shorter than 4 hops.

Chapter 5

PROTOTYPE

In order to verify MASS-ANET in a real-world environment, we developed several prototypes on commodity devices currently on the market. In the first Section of this chapter we introduce the specifications and the operating system of the smartphones used as prototypes. In section 5.2 we present a high-level description of the software. The central server application is reported in Section 5.3.

5.1 Smartphones

Commodity smartphones are used as prototypes in this research. The two low-cost adopted devices are the ZTE Blade and the Sony-Ericsson Xperia X8 (Figure 5.1). Both are based on Android (Section 5.1.1) and share the same chipset, with a 600MHz processor, and connectivity features (3G and Wi-Fi 802.11 b/g). A complete specification list is provided in Table 5.1.

(a) (b)

Figure 5.1: Smartphones used as prototypes: (a) ZTE Blade (b) Sony-Ericsson Xperia X8.

Feature	**ZTE Blade**	**Sony Ericsson Xperia X8**
2G Network	GSM 900 / 1800 / 1900	GSM 850 / 900 / 1800 / 1900
3G Network	HSDPA 900 / 2100	HSDPA 850 / 900 / 1900 / 2100
Dimensions	116 x 56.5 x 11.8 mm	99 x 54 x 15 mm
Weight	110g	104g
Display	480x800px, 3.5in, 256k colors	320x480px, 3.0in, 16M colors
Memory	150 MB storage, 512 MB RAM, 512 MB ROM	128 MB storage, 168 MB RAM
Card slot	microSD, up to 32GB	microSD, up to 16GB
GPRS	Yes	
EDGE	Yes	
3G Data	HSDPA, 7.2 Mbps; HSUPA, 5.76 Mbps	HSDPA, 7.2 Mbps; HSUPA, 2 Mbps
WLAN	Wi-Fi 802.11 b/g	
Bluetooth	Yes, v2.1 with A2DP, EDR	Yes, v2.1 with A2DP
OS	Android 2.3 custom	
CPU	600 MHz ARM 11 processor, Adreno 200 GPU, Qualcomm MSM7227 chipset	
GPS	Yes, with A-GPS support	
Battery	Standard battery, Li-Ion 1250 mAh	Standard battery, Li-Po 1200 mAh

Table 5.1: Smartphone specification

5.1.1 Android

Android is the Operating System (OS) installed on the prototype smartphones used to validate the results of this research thesis. Android, although considered an OS, is a software stack for mobile devices that includes an operating system, middleware and key applications [18]. It is based on the Linux kernel and its applications are executed by the Dalvik virtual machine. Dalvik is an adaptation of the Java virtual machine for mobile devices that allows the programmers to use the Java language. However, the low-level language C, which directly interacts with the Linux kernel, can still be used.

5.2 Prototype Implementation - Smartphones

In the following sub-section, an insight into the smartphone application implementation is given. Besides the entry point class to handle the start-up functions, the software is composed of several packages dealing with different tasks. Table 5.2 provides a brief description of the most important packages.

NetworkServices	Handles Wi-Fi peer-to-peer connections
CDS	Handles the CDS algorithm status
TransportService	Handles source-destination communications
CentralServer	Handles 3G communications with central server
Voice	Handles in/out voice
GUI	All the interfaces of the application
Location	Handles the GPS. Only for log purpose
Log	Handles the Log system
SpeedTest	Package used to test the links

Table 5.2: Application packages

5.2.1 Wi-Fi Ad-Hoc Mode

Although the most logical approach to build the ad-hoc network, with the Android prototypes, is that of using the Wi-Fi ad-hoc mode described in Section 2.2.3, this approach turned out to be very difficult in practice. In fact the ad-hoc mode is unsupported on Android, so no APIs are currently available. On the google website there is a three year old "open issue" [19] where thousands of users have left comments to request this feature since Jan 2008.

The device not only needs to "see" ad-hoc networks, but also to create them. Different approaches were used for this purpose, as for example OLSR (Section 2.1.2) described on its website [2], without any success. Eventually, we found an open-source Wi-Fi tether application for Android [45]. Tethering is the technique of sharing the internet connection, either via wireless (bluetooth or Wi-Fi) or via cable, from a mobile device, such as a smartphone, to other devices. If the internet connection is shared through Wi-Fi, an ad-hoc network is usually used and this is the case of the found application.

The application is called "Barnacle WiFi Tether" [45] and allows to share its 3G internet data connection by creating an ad-hoc Wi-Fi network to which other clients' devices can connect. Part of the Barnacle source code interacts, at low level, directly with the Linux Wi-Fi driver in order to enable the ad-hoc mode. Using Barnacle as a skeleton for this project, it was possible to use the ad-hoc Wi-Fi mode and build a prototype.

Each device starts its own ad-hoc network using the same SSID and a static distinct IP address. The network stays alive even if disconnected. The devices are able to see one another as soon as they are in range.

5.2.2 Peer-to-Peer Wi-Fi Communication

We used the User Datagram Protocol (UDP) [25] to exchange all types of packets, called datagrams, with the neighbour nodes in range. UDP, part of the Internet Protocol Suite, is a light weight protocol because it is state-less (i.e., connection-less), and it only supports error checking at the reception of a packet. UDP does not guarantee reliability, congestion control nor flow control.

Beacon messages are among the types of packets exchanged through UDP. These messages are broadcasted with a frequency of 1Hz (i.e., one per second). Before the beacon is sent, the CDS package is contacted to retrieve the current device status in the CDS network and any D_INFO messages to be included in its payload.

Even if UDP is really suitable to broadcast beacon messages, it is not a reliable protocol for important point-to-point packet transmissions. In MASS-ANET link set-up or tear down messages are critical. Therefore, we implemented a retransmission system in order to prevent the loss of critical data packets. The destination node of a packet sends an acknowledgement (ACK) message notifying the correct reception to the source. If the source does not receive the ACK within a certain interval of time, the packet is retransmitted. The retransmission is cut off after a certain amount of time (5 seconds) elapsed since the first transmission.

Wi-Fi link reliability

Beacons are sent by the nodes in order to establish links with neighbour devices and construct a list of them. The experiments showed beacon reception between devices far away from the nominal maximum distance of 100m. In an open field, a detection of few beacons was even registered at a distance of about 400m. Links at such a long range are unreliable and most of the transmitted packets are lost. Therefore, links cannot be automatically established based only on beacon reception.

In a reliable link, the received beacons are close to 100%. In order to establish connections only on reliable links, we implemented a connection hysteresis[1] based on beacon reception and loss rate within a time window.

The hysteresis time windows induce a delay in link detection. Therefore, tuning the right hysteresis values is a trade-off between reliability and link detection delay. On the one hand, the window should not be too long, otherwise the mobility poses a problem; on the other hand, it should not

---

[1] Hysteresis, from greek *hystérēsis*, “deficiency”, is the feature of a system where the output is determined not only by the input applied but also by its previous state [36].

be too short, because links connectivity could oscillate in presence of not completely reliable links.

Two different hystereses are considered. The first hysteresis is used to discriminate, with strict conditions, between reliable and unreliable links in order to establish new connections. The second hysteresis, with relaxed constraints, is used to check connected links and, if necessary, disconnect from them. The detailed hysteresis descriptions are:

1. is used by a node to establish a new connection with a neighbour. The beacon frequency used is 1Hz. The link is considered reliable and connected if 20 beacons are received out of 21 seconds, the 95%.
2. is used to check the reliability of a connected link between two nodes. If the received beacons are less than 9 out of 15 seconds, or 60%, the link is considered disconnected and the first hysteresis is applied again to re-establish a connection.

Asymmetric links can create problems in the network. Therefore, each node inserts into its beacons also the connected neighbours that pass the hysteresis process. A full duplex link is considered connected when it is connected at both sides. Moreover, redundant “coarse-grained” asymmetric paths are also taken into account in the central server between dominators. Therefore, end-to-end connections cannot be set up without reliable end-to-end connections in both ways.

5.2.3 Central Server - Device Communication

In the communication with the central server we used the Transmission Control Protocol (TCP) [52, 26]. TCP can guarantee a reliable transmission implementing flow control, congestion control and packets retransmission with reordering.

5.2.4 Multi-Hop Communication

TransportService

The TransportService package handles multi-hop connections such as voice calls between two non-neighbour devices. Therefore, the main function of these classes is to correctly forward received packets to the following hop. In case the node is a dominator, this package deals also with the forwarding tables sent by the central server.

Example application - Voice management

The use case implemented as proof of concept to validate this research is a real-time voice data transmission, a phone call, between two devices. In order to make phone calls, the voice needs to be recorded and reproduced. The input thread records a sound, and sends a voice packet every 190ms, using the Pulse-Code Modulation (PCM) [5] with 16 Bit encoding. This method produces uncompressed data, resulting in a packet payload of 4096 Bytes. The destination output thread also runs every 190ms to reproduce the voice packets received. A small voice packet buffer is used at the destination node to handle the variable inter-arrival packet delays introduced by the network allowing a smooth voice playback.

Initially, we implemented the retransmission technique also for the voice datagrams with the help of buffers on each node to reorder the packets. The drawback of this approach is the reduction of the total bandwidth available per link, and the introduction of an end-to-end delay of a few seconds in transmissions with more than six hops. During the experiments, it turned out that, with the use of the hysteresis introduced in Section 5.2.2, the established links are reliable enough to let us exclude the voice packets retransmission method. Moreover, most of the phone calls did not lose quality, and the only delay then associated was that produced by the forwarding computation and transmission, as shown by the experiment results. Additional technique to improve the calls' quality further, but out of the scope of this thesis, is the compression of the voice packets that could allow the use of packet retransmissions without the consumption of too much bandwidth.

5.3 Prototype Implementation - Central Server

The implementation of the central server was a less complex problem. Its core is composed of three main parts:

- The first, main component handles all the TCP connections and disconnections of the devices using multiple threads. All the online devices are kept in a list and a two-way communication, between them and the server, is always available through 3G.
- The second package handles the connectivity information about the dominator nodes. It creates, and constantly updates, a direct unweighed graph between each dominator. The open-source Java Universal Network/Graph (JUNG)[24] framework is used to show, in real-time, the graph connectivity of the network, see Figure 5.2.
- The last component handles the connection requests between the devices. It deals with computing the shortest path in the graph and updating the forwarding tables of the dominators on the selected path.

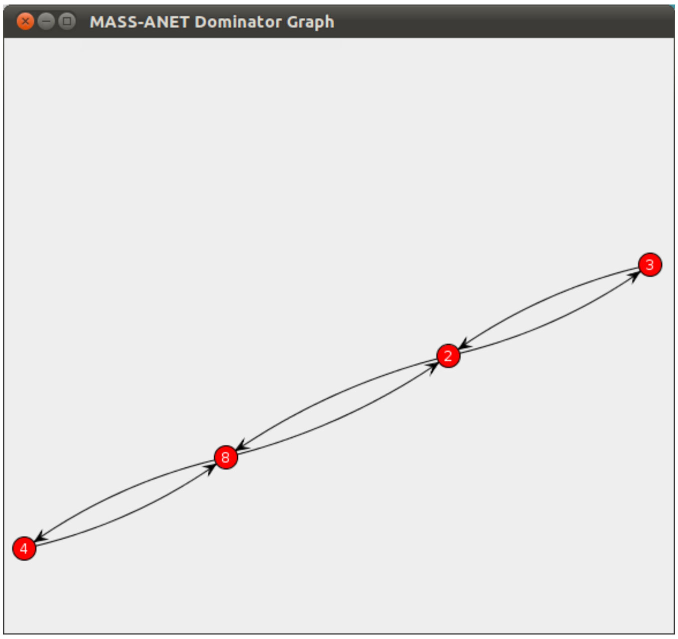


Figure 5.2: Screenshot of the real-time server dominator network graph during one of the experiments.

Chapter 6

EXPERIMENTAL RESULTS

In this chapter we evaluate the performance of MASS-ANET in a real world environment. In the first series of tests we analyse the maintenance phase in a network with simulated mobility. Then, the devices were deployed in several static locations for phone call tests.

6.1 Configuration

The mobility of the network used to test the maintenance phase was simulated in software. All seven devices were deployed on a table. Their reception capabilities were altered every 3 minutes to simulate the devices' mobility. Four different network topologies, shown in Figure 6.1, were interchanged instantaneously to simulate movement. The exchanged traffic in the network mobility experiments includes only MASS-ANET maintenance messages.

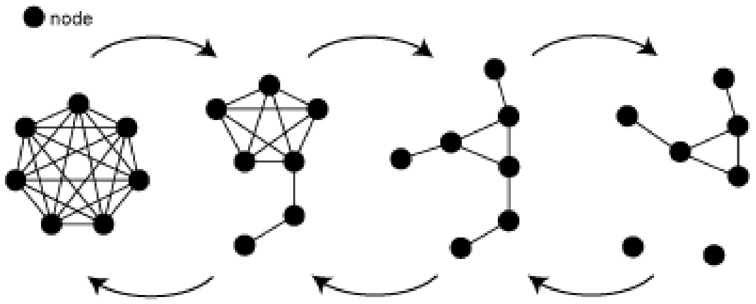


Figure 6.1: Network topologies interchanged in the mobility case scenario.

In the phone call experiments the network topology is a static chain of seven nodes. We chose the chain topology because it is an extreme case scenario with only one possible path between the edges. The experiments took place in the building of the Electrical Engineering, Mathematics and Computer Science faculty of Delft University of Technology. The chain of smartphones was formed in two ways. In the first series of experiments the devices were spread across the odd floors from the fifth to the seventeenth floor as shown in Figure 6.2. In the second, the devices were deployed inside offices on the 9th floor. In both experiments phone calls were made by the outers devices of the chain.

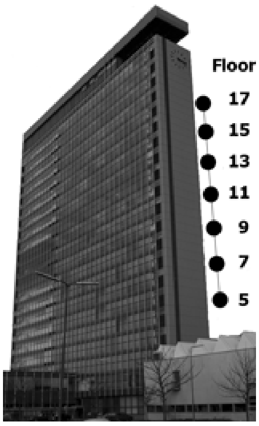


Figure 6.2: One of the experiment configurations in the TUDelft EWI building.

An example of the CDS structure from one of the chain experiments is shown in Figure 6.3.

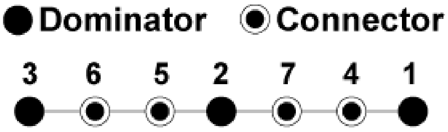


Figure 6.3: CDS structure of one of the experiments.

6.2 Results

A run of the maintenance phase tests in simulated mobility lasted for 45 minutes. The total 3G traffic exchanged was 198 bytes. It includes packets sent by dominators to the central server to notify changes in the dominator connectivity list. On average 2 bytes were sent every 3 minutes per node. Even if the network was mobile, the 3G traffic was quite low because the links were completely reliable. In fact, the smartphones were next to each other, so the CDS network did not fluctuate during the steady periods of the nodes. The Wi-Fi control messages used to handle the CDS structure amounted to 2.2KB.

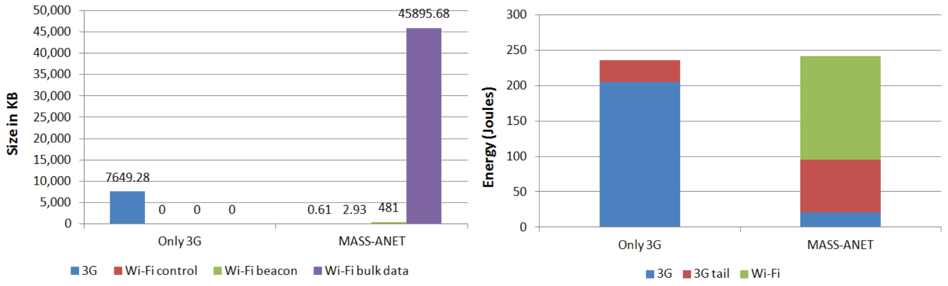
A run of the static phone call experiments was set up to analyse the amount of 3G traffic offloaded to Wi-Fi. The smartphones were deployed in several offices forming a chain. The run lasted 55 minutes. The edges of the chain made two phone calls of 6 and 9 minutes respectively. The total voice traffic offloaded to the Wi-Fi network amounted to 7.47MB. The overall 3G traffic transferred during the run was only 624 bytes. 332 bytes for MASS-ANET maintenance and 292 bytes for phone call set-up and routing. The Wi-Fi control messages amounted to 2.93KB and the Wi-Fi beacons to 481KB.

In a static network with completely reliable links 3G messages are exchanged only at the bootstrap phase to set-up the CDS network. Therefore, the total 3G traffic is constant over time. However, the traffic registered during the run of the static experiment was even higher than the run with network mobility. In fact, the CDS network in the static run fluctuated several times during the 55 minutes test because the links were not totally reliable. Thus, the devices were deployed at the limits of their radio ranges inside several offices.

Despite the network fluctuations MASS-ANET offloaded 99.99% of the 3G traffic. Figure 6.4 (a) shows a comparison between “only 3G” and MASS-ANET on traffic size. The difference of 3G traffic between the two solutions is 4 orders of magnitude (7.47MB vs 0.6KB). Moreover, the high Wi-Fi bulk data traffic is due to the size of the original voice traffic multiplied by 6 hops along the 7 nodes chain. However, Wi-Fi is cheaper than 3G (in terms of both energy consumption and money) so it can be exploited without too many restrictions.

Figure 6.4 (b) shows a comparison between “only 3G” and MASS-ANET on an approximation of the energy consumption according to the model of [39]. The average MASS-ANET energy consumption per node is comparable to the energy drained by the 3G radio for the same download traffic.

The results of the traffic sizes are directly retrieved from the device logs. The estimation of the energy consumption was not so easy. In the “only 3G” case we applied the simple 3G energy consumption model of [39]. However, the model cannot be directly applied to estimate the energy consumption of MASS-ANET because the model assumes a traffic burst of at least 1KB while MASS-ANET exchanges only a few bytes at a time. We investigate the time between consecutive 3G transmissions on each node to compute the time the nodes were affected by tail energy. The total time of tail energy mode on all the nodes is so multiplied by 0.62 Joules/second [39].



(a) (b)

Figure 6.4: Experiment comparison results between only 3G and MASS-ANET: (a) total network traffic (b) energy consumption per node.

To compute the 3G transmission energy, we multiplied the number of 3G packets sent by the network by 3 Joule according to [39]. The Wi-Fi energy consumption is split in maintenance energy and transmission energy. The maintenance energy is computed recording the time the Wi-Fi radio was active in each device. Moreover, the exchanged Wi-Fi data size was used to compute the energy consumption due to transmissions.

Several Quality of Service metrics relating to the phone call experiments are now discussed. The first metric used to describe the link reliability is the packet reception rate. The reception rate depends on several factors such as the location of the smartphones. The end-to-end packet reception rate was between 67% and 93%.

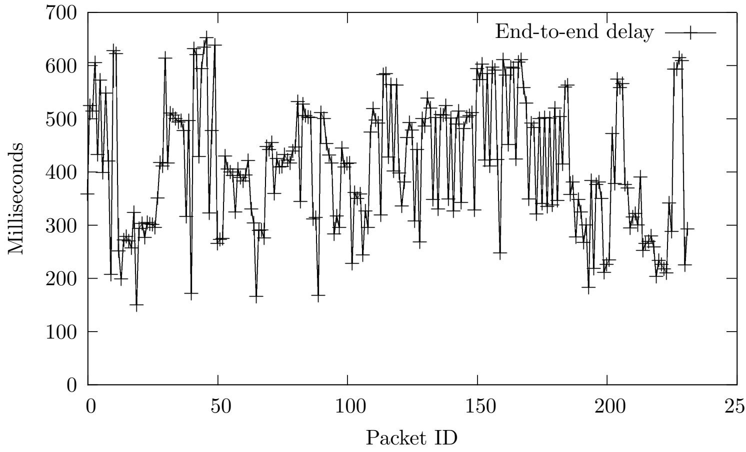


Figure 6.5: Example of end-to-end packet delay in a chain of seven nodes.

The average end-to-end packet delay across all our experiments is about

450ms with values between 150ms and 850ms. End-to-end packet delay gives an overview of the behaviour of the traffic in the network. The delay knowledge is useful for determining buffer sizes and retransmission policies. The behaviour of the delay in one of the experiments is shown in Figure 6.5. The measured delay is an approximation because the timing of the logged events is unsynchronized across the phones, so a synchronization process is required. Each packet sent has a unique ID. We assume that a packet is sent or received when this event is recorded in the log file of the node. We retrieve from the log files the difference in time between sent packets from a node and the same packets, through their ID, received by the second node. This difference is used to synchronize the clock of the second node. This operation is repeated for all pairs of nodes in the chain.

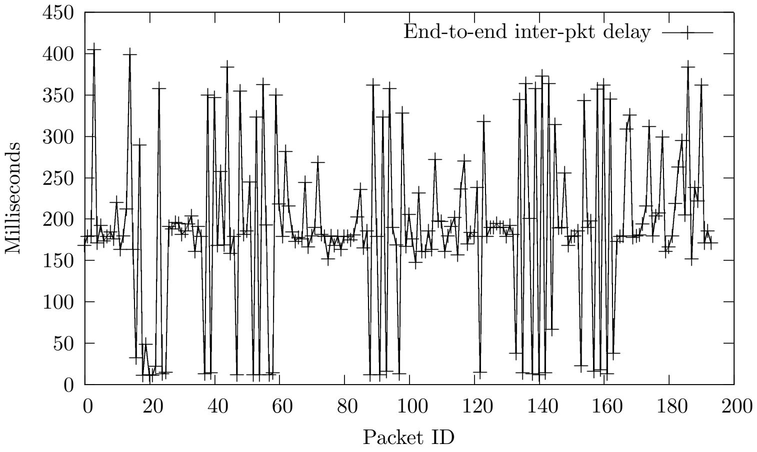


Figure 6.6: Example of received end-to-end inter-packet delay.

Another important measurement is the inter-packet delay or the delay between consecutive received packets at the end of the seven node chain. Inter-packet delay is important especially for real time traffic such as phone calls. If the delay is too high the voice cannot be reproduced smoothly. Therefore, the chain should be shortened. The inter-packet delay measured in the experiments is on average 190ms with values between 11ms and 424ms. Figure 6.6 shows an example of the inter-packet delay from one of the runs. The shape of the graph suggests that packets are often sent in burst of two. A possible cause could be the Wi-Fi channel contention time. Once a node wins the contention, it will send all the packets stored to be forwarded, which seems to be two in this case.

Through these sets of experiments we showed that MASS-ANET can be used in practice, even for real-time traffic, and can accomplish its primary goal of offloading traffic from the 3G networks.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

Mobile devices with internet capabilities, such as smartphones and tablets, are becoming part of everybody's daily-life. More and more people want to access the internet everywhere using 3G data technology. This trend is expected to grow at an amazing rate [9]. However, the 3G network capacity is already pushed towards the limits and the current infrastructure, alone, will not be able to sustain such an increase of demand.

Different research projects as well as telecommunication companies try to offload the 3G data traffic to Wi-Fi access points (APs) [43][53]. However, APs have a limited range and do not offer a feasible solution for providing coverage for entire cities.

The goal of this thesis is to offload substantially more 3G internet traffic by increasing the APs range without any need of deploying and installing new infrastructure. This is achieved by proposing a new peer-to-peer ad-hoc network technique, called "Mobile Assisted Ad-hoc Networking" (MASS-ANET). Both the devices and the APs can exploit it through a simple software update. A smartphone can connect to an "out of range" AP through a chain of other devices. Besides internet access, this solution also allows exchange of data or real-time voice traffic (i.e., phone calls) between the devices in MASS-ANET.

MASS-ANET requires the availability of a short and a long range communication system. Two overlay networks and two routing mechanisms are exploited by MASS-ANET. A short range link (e.g., Wi-Fi) is used for neighbour discovery and to set-up the first overlay network. This overlay network is built using a Connected Dominating Set (CDS) algorithm. The CDS nodes can route packets to maximum 3-hop neighbours. A Maximal

Independent Set (MIS) is used as the base for the CDS.

A long range link (e.g., 3G) is used to communicate with a central server that assists global node discovery and routing. The nodes notify their presence to the server when they switch on/off to accomplish part of the node discovery functionality. Beside that, only a subset of nodes of the CDS structure, the MIS nodes, are allowed to use the long range link. The server keeps track of the second overlay network based on the MIS node connectivity. Therefore, the second routing mechanism is exploited by the server that computes end-to-end shortest paths in the network through MIS nodes. Since the network is mobile, MIS nodes constantly update their connectivity to the server. Moreover, the server keeps computing shortest paths and sends forwarding tables to MIS nodes. Therefore, MIS nodes, together with the CDS routing information, are able to correctly route packets.

The final design and implementation of MASS-ANET is evaluated in simulations using different application scenarios and mobility parameters. In the example of a single download described in Chapter 4.5 MASS-ANET offloaded 96.2% of the data traffic from the 3G network to the Wi-Fi network. The average energy consumption per node, using 3G network as the global channel, is 36% of the energy used by the 3G radio for the same download traffic. Moreover, MASS-ANET does not require high bandwidth on the global channel because only few bytes are exchanged. Therefore, the 3G technology can be replaced by the slower but cheaper, in terms of energy consumption, GSM. The average energy consumption per node using GSM is only 14% of the energy used by the 3G radio for the same download traffic.

To prove the validity of this research, several commodity devices have been used as prototypes. The successful set-up and forwarding of phone calls in a real-world environment shows the feasibility of this new approach and confirms that it could become reality right now with just a software update. In the static chain of seven node experiment described in Chapter 6 MASS-ANET offloaded the 99.99% of 3G data traffic to the Wi-Fi network. The average MASS-ANET energy consumption per node was comparable to the energy consumption of a 3G radio for the same download traffic.

This research focused on offloading 3G data to Wi-Fi networks. However, MASS-ANET can be applied to any kind of mobile network where the nodes can use both local and long range communication channels to contact a central server. This could be the case, for example, of military applications. In fact soldiers in a battlefield could be connected to one another with a short range link while using a temporary WiMax or LTE channel, as global one, to assist that network.

7.2 Future Work

Although the result of this thesis is a fully functional solution, it is still a proof of concept. Different features can be developed to improve its performance and usability.

- Up to the present, the ad-hoc network includes only Android devices. Therefore, the next step would be to implement the MASS-ANET protocol also on internet Wi-Fi access point to give the real internet capabilities to the network.

- In our simulation scenarios MASS-ANET is maintained active in all the nodes in the network. However, MASS-ANET could be studied using another prospective. MASS-ANET can be activated only on a small subset of the network around a requesting node. In fact, the nodes can be completely quiet and just listen to the Wi-Fi channel. The beacons' activity can be used as start-up signal with a Time-To-Live from the requesting node. Therefore, only a limited number of nodes will be activated around the requesting node.

- Several techniques can be explored to further reduce the number of devices that need to constantly exchange messages with the server through 3G. During the design phase of this project, the Beacon Vector Routing (BVR) technique (Section 2.1.4) was considered. Preliminary simulations show that implementing the BVR protocol on top of the dominator network, with a "Beacon" every 3-hop, can reduce the average number of dominators from 60 to about 12, within 1Km^2 and range radius of 100m. However, a deeper study is required because the network performance in mobile network can be seriously affected.

- The current network graph implementation in the central server is directed and unweighed. Flow and congestion control, in the Wi-Fi network, can be implemented using a weighed graph representation and adding few bytes of information to the 3G messages from the dominators to the server. Then, Quality of Service (QoS) techniques can be easily integrated in the network.

- The IP addresses of the Wi-Fi network assigned to the prototype devices are static. In a real deployment of this project, the central server should act as Dynamic Host Configuration Protocol (DHCP) for a dynamic IP allocation. The geographic position of the 3G tower, which the devices are connected to, can help to limit the range of addresses.

- No security policies are considered in this proof of concept. All the Wi-Fi messages are exchanged in the clear. Encryption techniques can be applied to those messages. The 3G connection can be used by the two end of the communication, to agree on the encryption methodology and/or to exchange the key to decode the messages.

Bibliography

[1] Wi-Fi Alliance. Wi-Fi certified$^{\mathrm{TM}}$n: Longer-range, faster-throughput, multimedia-grade Wi-Fi® networks. Sept 2009.

[2] OLSRD an adhoc wireless mesh routing daemon. Getting ad-hoc mode on android. http://www.olsr.org/?q=olsr_on_android.

[3] A.Neumann, C.Aichele, and M.Linder. Better Approach To Mobile Ad-hoc Networking (B.A.T.M.A.N.). IETF Draft, Oct 2008.

[4] B.Das and V.Bharghavan. Routing in ad-hoc networks using minimum connected dominating sets. In *IEEE International Conference on Communications*, volume 1 of *ICC '97*, pages 376–380, Jun 1997.

[5] B.M.Oliver and C.Shannon. Communication system employing pulse code modulation. U.S. Patent 2,801,281, granted in 1956. Original study of Alec Reeves (1939).

[6] B.N.Clark, C.J.Colbourn, and D.S.Johnson. Unit disk graphs. In *Discrete Mathematics*, volume 86, pages 165–177, Dec 1990.

[7] C.Bettstetter, G.Resta, and P.Santi. The node distribution of the random waypoint mobility model for wireless ad hoc networks. In *IEEE Transactions on Mobile Computing*, volume 2, pages 257–269, Jul 2003.

[8] C.E.Perkins and E.M.Royer. Ad-hoc on-demand distance vector routing. In *Proceedings of the Second IEEE Workshop on Mobile Computer Systems and Applications*, WMCSA '99, pages 90–100, New Orleans, LA, Feb 1999. IEEE Computer Society.

[9] Cisco. Cisco visual networking index: Global mobile data traffic forecast update, 2010-2015. 2011.

[10] D.B.Johnson and D.A.Maltz. Dynamic source routing in ad hoc wireless networks. In *Mobile Computing*, pages 153–181, 1996.

[11] DiscussTech.org. Difference between 2G and 3G. http://discusstech.org/2011/07/difference-between-2g-and-3g/, July 2011.

[12] D.Murray, M.W.Dixon, and T.Koziniec. An experimental comparison of routing protocols in multi hop ad hoc networks. In *Australasian Telecommunication Networks and Applications Conference*, ATNAC '10, pages 159–164. Auckland, New Zealand, Nov 2010.

[13] D.Zhou, M.T.Sun, and T.H.Lai. A timer-based protocol for connected dominating set construction in ieee 802.11 multihop mobile ad hoc networks. In *Proceedings of the International Symposium on Applications and the Internet*, SAINT '05, pages 2–8. IEEE Computer Society, Jan 2005.

[14] E.Kranakis, H.Singh, and J.Urrutia. Compass routing on geometric networks. In *11th Canadian Conference on Computational Geometry*, CCCG '90, pages 51–54, Aug 1999.

[15] E.Sampathkumar and H.B.Walikar. The connected domination number of a graph. In *J.Math.Phys.Sci.*, volume 13, pages 607–613, 1979.

[16] F.Dai and J.Wu. An extended localized algorithm for connected dominating set formation in ad hoc wireless networks. In *IEEE Transaction on Parallel and Distributes System Conference on Computer Communications*, volume 15, pages 908–920, Oct 2004.

[17] G.Anastasi, E.Borgia, M.Conti, and E.Gregori. Wi-Fi in Ad Hoc Mode: A measurement study. In *Proceedings of the Second IEEE International Conference on Pervasive Computing and Communications*, PERCOM '04, pages 145–154. IEEE Computer Society, 2004.

[18] Google. What is android? http://developer.android.com/guide/basics/what-is-android.html.

[19] Google, Android - Issues. Wi-Fi, support ad hoc networking open issue. http://code.google.com/p/android/issues/detail?id=82.

[20] H.Frey. Scalable geographic routing algorithms for wireless ad hoc networks. In *IEEE Network Mag.*, pages 18–22. IEEE Communications Society, July/Aug 2004.

[21] I.D.Chakeres and C.E.Perkins. Dynamic MANET on demand (DYMO) routing protocol. IETF Draft, 2006.

[22] J.Chroboczek. The Babel Routing Protocol. IETF RFC 6126, Apr 2011.

[23] J.Lin, F.Liu, T.Korakis, Z.Tao, E.Erkip, and S.Panwar. A WiFi measurement in 802.11g networks. Technical Note. Available online: http://wireless.poly.edu/attachment/wiki/WikiStart/HTmeasurement.pdf.

[24] J.Madadhain, D.Fisher, P.Smyth, S.White, and Y.B.Boey. Analysis and visualization of network data using JUNG. *Journal of Statistical Software*, 10:1–35, 2005.

[25] J.Postel. User Datagram Protocol (UPD). IETF RFC 768, Aug 1980.

[26] J.Postel. Transmission Control Protocol (TCP). IETF RFC 793, Sep 1981.

[27] J.Wu and H.Li. On calculating connected dominating set for efficient routing in ad hoc wireless networks. In *Proceedings of the 3rd international workshop on Discrete algorithms and methods for mobile computing and communications*, DIALM '99, pages 7–14. ACM New York, NY, USA, 1999.

[28] J.Yoon, M.Liu, and B.Noble. Random waypoint considered harmful. In *Proceedings of the 22nd IEEE International Conference on Computer Communications*, volume 2 of *INFOCOM '03*, pages 1312–1321, Apr 2003.

[29] K.M.Alzoubi, P.J.Wan, and O.Frieder. Message-optimal connected dominating sets in mobile ad hoc networks. In *Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing*, MobiHoc '02, pages 157–164. ACM New York, NY, USA, May 2002.

[30] K.Sakai, F.Shen, K.M.Kim, M.T.Sun, and H.Okada. Multi-initiator connected dominating set construction for mobile ad hoc networks. In *IEEE International Conference on Communications*, ICC '08, pages 2431–2436, May 2008.

[31] K.Sakai, M.T.Sun, and W.S.Ku. Fast connected dominating set construction in mobile ad hoc networks. In *IEEE International Conference on Communications*, ICC '09, pages 1–6, Jun 2009.

[32] K.Sakai, M.T.Sun, W.S.Ku, and H.Okada. Maintaining cds in mobile ad hoc networks. In *Proceedings of the 3rd International Conference on Wireless Algorithms, Systems, and Applications*, WASA '09, pages 141–153. Springer-Verlag, Berlin, DE, 2008.

[33] L.Zhang, B.Tiwana, Z.Qian, Z.Wang, R.P.Dick, Z.M.Mao, and L.Yang. Accurate online power estimation and automatic battery behavior based power model generation for smartphones. In *Proceedings of the eighth IEEE/ACM/IFIP international conference on Hardware/software codesign and system synthesis*, CODES/ISSS '10, pages 105–114. ACM New York, NY, USA, 2010.

[34] M.Abolhasan, B.Hagelstein, and J.C.-P.Wang. Real-world performance of current proactive multi-hop mesh protocols. In *Proceedings of the 15th Asia-Pacific conference on Communications*, APCC '09, pages 42–45. IEEE Press Piscataway, NJ, USA, 2009.

[35] M.K.Kjaergaard, J.Langdal, T.Godsk, and T.Toftkjaer. EnTracked: energy-efficient robust position tracking for mobile devices. In *Proceedings of the 7th international conference on Mobile systems, applications, and services*, MobiSys '09, pages 221–234. ACM New York, NY, USA, 2009.

[36] M.Krasnosel'skii and A.Pokrovskii. *Systems with Hysteresis.* Springer-Verlag, New York, 1989.

[37] M.Linder and A.Neumann. BATMAN (Better Approach To Mobile Ad Hoc Networking). http://www.open-mesh.org.

[38] M.Min, F.Wang, D.Z.Du, and P.M.Pardalos. A reliable virtual backbone scheme in mobile ad-hoc networks. In *IEEE International Conference on Mobile Ad-hoc and Sensor Systems*, MASS '04, pages 60–69, Oct 2004.

[39] N.Balasubramanian, A.Balasubramanian, and A.Venkataramani. Energy consumption in mobile phones: a measurement study and implications for network applications. In *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference*, IMC '09, pages 280–293. ACM New York, NY, USA, 2009.

[40] P.Bose, P.Morin, I.Stojmenovic, and J.Urrutia. Routing with guaranteed delivery in ad-hoc wireless networks. In *3rd Int. Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications*, DIAL-M '99, pages 48–55, 1999.

[41] P.Gardner-Stephen. The serval project: Practical wireless ad-hoc mobile telecommunications. http://www.servalproject.org/.

[42] P.J.Wan, K.M.Alzoubi, and O.Frieder. Distributed construction of connected dominating set in wireless ad hoc networks. In *Proceedings of the 21st IEEE International Conference on Computer Communications*, volume 3 of *INFOCOM '02*, pages 1597–1604, 2002.

[43] Qualcomm. 3G-Wi-Fi smart mobility - helping operators maximize data capacity. http://www.qualcomm.com/research/wi-fi-offload.

[44] R.M.Karp and A.Wigderson. A fast parallel algorithm for the maximal independent set problem. In *Proceedings of the 16th annual ACM symposium on Theory of computing*, STOC '84, pages 266–272. ACM New York, NY, USA, 1984.

[45] S.Jakubczak. Barnacle Wi-Fi Tether. http://szym.net/barnacle/, Author home page - http://people.csail.mit.edu/szym/.

[46] S.Jakubczak. Dythr - local headline dissemination. http://szym.net/dythr/.

[47] S.Ratnasamy, B.Karp, S.Shenker, D.Estrin, R.Gvoindan, L.Yin, and F.Yu. Data-centric storage in sensornets with ght, a geographic hash table. In *Mobile Networks and Applications*, volume 8, pages 427–442, 2003.

[48] S.Tisue and U.Wilensky. NetLogo: A simple environment for modeling complexity. In *International Conference on Complex Systems*, pages 16–21, 2004.

[49] T.Clausen and P.Jacquet. Optimized Link State Routing Protocol (OLSR). IETF RFC 3626, Oct 2003.

[50] T.Wilhelm. Auto-BAHN. http://hackerdemia.com/.

[51] U.C.Kozat, G.Kondylis, B.Ryu, and M.K.Marina. Virtual dynamic backbone for mobile ad hoc networks. In *IEEE International Conference on Communications*, volume 1 of *ICC '01*, pages 250–255, Jun 2001.

[52] V.G.Cerf and R.E.Kahn. A protocol for packet network intercommunication. In *IEEE Transactions on Communications*, volume 22, pages 637–648, May 1974.

[53] W.Song, W.Zhuang, and A.Saleh. Interworking of 3G cellular networks and wireless LANs. In *Int. J. Wireless and Mobile Computing*, volume 2, pages 237–247. Inderscience Publishers, Geneva, CH, 2007.