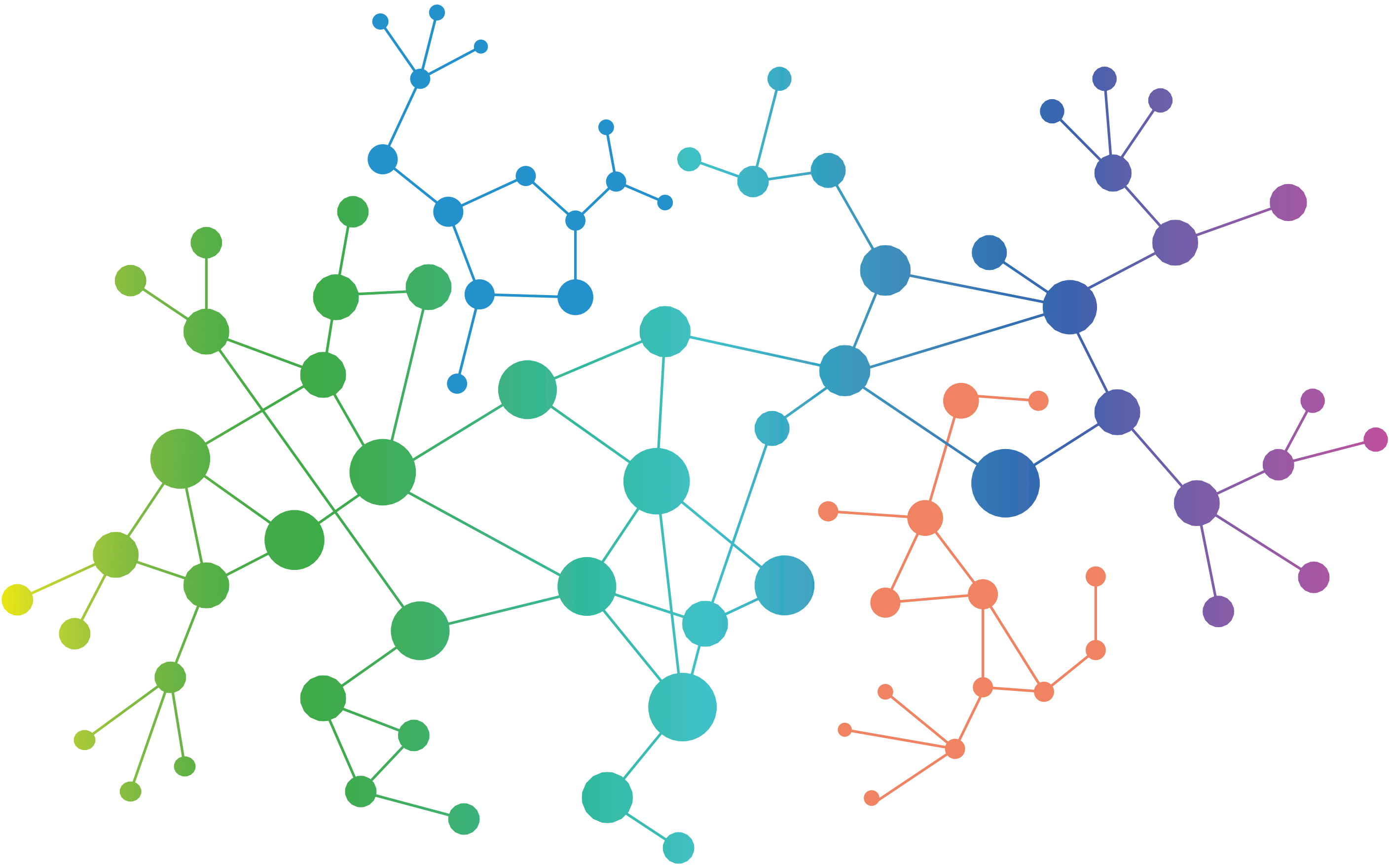


Luís Alexandre Morais da Conceição

Context-aware Clustering in Mobile Ad-hoc Networks

Tese de Doutoramento do Programa de Doutoramento em Ciências e Tecnologias da Informação, orientada pela Professora Doutora Marília Pascoal Curado e apresentada ao Departamento de Engenharia Informática da Faculdade de Ciências e Tecnologia da Universidade de Coimbra

Maio de 2016

• U C •

Universidade de Coimbra

Department of Informatics Engineering
Faculty of Sciences and Technology
University of Coimbra

Context-aware Clustering in Mobile Ad-hoc Networks

Luís Alexandre Morais da Conceição

Doctoral Program in Information Science and Technology
PhD Thesis submitted to the University of Coimbra

Advised by Prof. Dr. Marília Pascoal Curado

May 2016

Departamento de Engenharia Informática
Faculdade de Ciências e Tecnologia
Universidade de Coimbra

Agrupamento em Redes Móveis Ad-hoc com base em Informação de Contexto

Luís Alexandre Morais da Conceição

Programa de Doutoramento em Ciências e Tecnologias da Informação
Tese de Doutoramento apresentada à Universidade de Coimbra

Orientado pela Prof. Dra. Marília Pascoal Curado

May 2016

This work was partially supported by Fundação para a Ciência e Tecnologia (FCT) under the grant SFRH/BD/81829/2011.

Acknowledgements

I would like to express my adamant thanks and deepest gratitude to my advisor Marilia Curado. For more than ten years you have been a teacher, a mentor and a friend, whose motivation and encouragement truly made a difference, not only in my work, but also in my life. For all the persistent meetings, reviews, discussions and advices a profound thank you.

I will always remember with joy and warmth the time spent at the Laboratory of Communications and Telematics. Strong relationships have arisen under a dome of hard work, arguments, disputes, pranks and many sleepless nights. Thank you all for the incredible support, in particular to Bruno Sousa, David Palma, João Almeida, João Gonçalves, Luís Cordeiro, Rui Vilão and Vitor Bernardo.

I would like to thank the Portuguese Foundation for Science and Technology for the PhD Scholarship (SFRH/BD/81829/2011) to the EU IST project MICIE (FP7-ICT-225353/2008) and to the national project MORFEU (PTDC/EEA-CRO/108348/2008). Additionally an acknowledgement to the OPNET University Program for providing the licenses required for the OPNET Modeler Wireless Suite R.

There are some things you don't have to acknowledge to known they're there. I would like to thank my Mother and Father, for the dedication, sacrifice and support they provided me through my entire life. Your admirable personality and moral conduct certainly shaped the person that I am today.

Last, but absolutely not the least, I would like to thank my wife Liliana and our daughter Maria. Liliana, you were always there during the most difficult moments. Thank you for your patience when I had to work weekends and holidays. Thank you for giving me a "reality check" during my most troubling moments. But above all, thank you for always believing in me, even when I didn't. To my daughter Maria, thank you for being a daily source of love and inspiration, helping daddy to aspire better purposes in life. To my Consort and Offspring I am eternally grateful.

Abstract

The increase of wireless communication devices is leading to a paradigm change in traditional network technologies, adapting from infrastructured and centralised toward distributed and self-organised. The Ad-hoc communication paradigm allows the deployment of autonomous and infrastructure-less networks organised in a distributed fashion. However, the continuous demand for larger and denser networks presents great challenges regarding scalability.

Clustering has been the subject of an extensive amount of proposals aiming to provide scalability to Mobile Ad-hoc Networks (MANETs). Indeed, the performance demands required by high mobility in large networks makes clustering a crucial solution to deliver an hierarchical topology in order to allow scalability. Regarding this issue, this work addresses the scalability of MANETs by proposing diverse clustering solutions capable of managing high mobile and large networks.

Addressing the most basic requirements of MANETs, which involve large networks and high mobility, the Smart and Balanced Clustering (SALSA) solution is presented first. In contrast with most existent clustering schemes, SALSA is a distributed clusterhead-free scheme, assigning equal management tasks to all nodes. This paradigm eliminates the overhead inherent to clusterhead re-elections and bandwidth bottlenecks caused by central management. In addition, SALSA uses a cluster balance mechanism promoting an even distribution of nodes among clusters. This solution also proposes a cluster quality metric to assign nodes to the most suitable clusters, regarding connectivity and free amount of positions within clusters. Results by simulation evaluation show a large percentage of clustered and stable nodes while maintaining a low overhead, even in large networks with high mobility.

With regards to the importance of location information to help the creation and management of clusters, the Clustering for Indoor and Dense Networks (CIDNET) and Adaptive and Location-aware Clustering (DiLoC) solutions were developed during this work. These schemes were motivated by the specific objective of providing scalability in dense indoor networks. In indoor environments, with the lack of better location positioning technologies (such as Global Positioning System), CIDNET uses existent Wireless Local Area Network (WLAN) infrastructures to provide location references in order to reduce the complexity of cluster management activities. In contrast with CIDNET, DiLoC also takes advantage of WLAN infrastructures but does not depend on them. Where and when no infrastructures are present, it uses special nodes, called anchor nodes, to serve as reference location points. Evaluation results demonstrate that a location-awareness clustering solution is capable of providing a more stable network topology, but with a light increase of overhead.

Social relationships information can also improve the performance of networks. The Social-aware Clustering Scheme (SoCS) is inspired by social similarity between nodes to improve cluster management decisions. People with similar interests, location or profession interact more often that strangers, thus the probability of being located in the same position is higher. Inspired by this fact, SoCS measures the connectivity frequency and durability between nodes to make better cluster assignment decisions. The obtained results show a significant increase of network scalability, improving the routing and traffic performances, particularly when a social mobility model is used.

This work also proposes a mobility model aiming to assess the network performance in post-disaster environments. The necessity of modelling MANETs in such environments, led to the development of Human Behaviour for Disaster Areas (HBDA) mobility model. The HBDA model mimics human behavioural trajectories during Search for Victim (SFV) operations in post-disaster environments. Results show the contrast of network performances between the usage of different mobility models.

Resumo

O crescimento de dispositivos de comunicação sem-fio tem dado origem a uma mudança no paradigma de tecnologias de rede tradicionais, substituindo redes centralizadas com base em infraestruturas por redes distribuídas e automaticamente organizadas. O paradigma de comunicação Ad-hoc possibilita a criação de redes autónomas sem-infraestrutura organizadas de maneira distribuída. Todavia, a existência de redes com maior dimensão e densidade apresenta um grande desafio no que respeita à escalabilidade.

O agrupamento tem sido foco de uma vasta gama de propostas no sentido de fornecer escalabilidade a redes móveis Ad-hoc. A necessidade de desempenho requerida pela alta mobilidade em redes de larga escala faz do agrupamento uma solução essencial para fornecer uma topologia hierárquica de modo a permitir escalabilidade. Face a esta questão, este trabalho aborda a escalabilidade de redes móveis Ad-hoc apresentando diversas soluções de agrupamento capazes de gerir redes de grande escala com alta mobilidade.

No sentido de fornecer os requisitos mínimos em redes móveis Ad-hoc, entre eles redes de grande escala e alta mobilidade, a solução de agrupamento inteligente e equilibrado (*Smart and Balanced Clustering* - SALSA) é apresentada em primeiro lugar. Contrariamente à maioria dos esquemas de agrupamento existentes, o SALSA é um esquema distribuído sem nós centrais, atribuindo tarefas de gestão semelhantes a todos os nós. Utilizando este paradigma é possível eliminar o custo de manutenção inerente à re-eleição de nós centrais e a congestão causada por uma manutenção central. Além disso, o SALSA utiliza um mecanismo de equilíbrio de grupos para promover uma distribuição uniforme de nós entre grupos. Esta solução também propõe uma métrica referente à qualidade de grupos de forma a escolher o grupo mais adequado para um nó, baseada na conectividade e posições disponíveis dentro dos grupos. Os resultados, obtidos através de uma avaliação por simulação, demonstram uma alta percentagem de nós agrupados e estáveis, conjuntamente com um baixo custo de manutenção, mesmo em redes grandes com alta mobilidade.

Face à importância que a informação de localização tem na criação e gestão de grupos, foram desenvolvidos os esquemas de agrupamento para redes interiores e densas (*Clustering for Indoor and Dense Networks* - CIDNET) e de agrupamento adaptável e baseado em localização (*Adaptive and Location-aware Clustering* - DiLoC). Os referidos esquemas foram desenvolvidos com o objectivo de fornecer escalabilidade em redes densas e de interior. Em espaços interiores, devido à falta de tecnologias de localização mais adequadas (tal como o *Global Positioning System (GPS)*), o CIDNET tira partido de infraestruturas de rede local sem fios existentes para fornecer pontos de referência por forma a reduzir a complexidade das operações de gestão de grupos. O esquema DiLoC também utiliza as referidas infraestruturas, no entanto não depende inteiramente delas. Quando não existem infraestruturas de rede local sem fios, dispõe de nós especiais, intitulados nós âncora, que servem como referência de localização. Os resultados de avaliação demonstram que um esquema de agrupamento, baseado em contexto de localização, é capaz de gerir uma topologia de rede mais estável, no entanto com um pequeno aumento do custo de manutenção.

A informação proveniente de relações sociais pode também favorecer o desempenho de redes. O esquema de agrupamento baseado na componente social (*Social-aware Clustering Scheme* - SoCS) baseia-se na semelhança social entre nós de forma a melhorar a gestão de grupos. Pessoas com interesses, localização ou profissão semelhantes tendem a interagir mais regularmente do que estranhos, portanto a probabilidade de se encontrarem na mesma área é maior. Baseado neste facto, a solução SoCS determina a frequência e durabilidade da conexão entre nós de forma a melhorar decisões de atribuição de grupos. Os resultados obtidos comprovam um aumento significativo de escalabilidade, favorecendo o desempenho de encaminhamento e de tráfego, principalmente quando avaliado utilizando um modelo de mobilidade social.

Este trabalho também propõe um modelo de mobilidade desenhado para avaliar o desempenho da rede em cenários pós-desastre. A necessidade de modelar redes móveis Ad-hoc nestes cenários levou ao desenvolvimento do modelo de mobilidade comportamento humano em áreas de desastre (*Human Behaviour in Disaster Areas* - HBDA). Este modelo gera trajectórias baseadas na modelação do comportamento humano durante operações de busca de vítimas em áreas pós-desastre. Os resultados realçam o contraste de desempenho entre diferentes modelos de mobilidade.

Foreword

The work described in this thesis was conducted at the Laboratory of Communications and Telematics of the Centre for Informatics and Systems of the University of Coimbra. Part of this work was developed in the context of the project MORFEU - Multi-Objective Robot Fleet for improved commUnication (PTDC/EEA-CRO/108348/2008).

Several contributions were published during the development of this thesis, listed as follows.

- Luís Conceição, Marilia Curado (2011). “Smart and balanced clustering for MANETs” in proceedings of the 10th International Conference on Ad Hoc Networks and Wireless.
- Luís Conceição, Marilia Curado (2012). “Clustering for Indoor and Dense MANETs” in the proceedings of the 16th International Conference on Next Generation Wired/Wireless Advanced Networks and Systems.
- Luís Conceição, Marilia Curado (2013). “Onto scalable wireless ad hoc networks: Adaptive and location-aware clustering” in the Journal Ad Hoc Networks, Elsevier. Volume 11, Issue 8, November 2013, Pages 2484-2499, ISSN 1570-8705, doi: 10.1016/j.adhoc.2013.07.001. Impact factor: 1.530.
- Luís Conceição, Marilia Curado (2013). “Modelling Mobility Based on Human Behaviour in Disaster Areas” in the proceedings of the 11th International Conference on Wired/Wireless Internet Communications.
- Luís Conceição, Marilia Curado (2014). “Modelling Mobility Based on Obstacle-Aware Human Behaviour in Disaster Areas” in the Journal Wireless Personal Communications, Springer. Volume 82, Issue 1, Pages 451-472, ISSN 0929-6212, doi: 10.1007/s11277-014-2235-8. Impact factor: 0.653.
- Luís Conceição, Marilia Curado (2016). “Social-aware Clustering for Wireless Ad hoc Networks” in the proceedings of the 8th Wireless Days International Conference.

Contents

List of Figures

List of Tables

List of Algorithms

Acronyms

3-hBAC	3-hop Between Adjacent Clusterheads
5G	5^{th} Generation
AODV	Ad-Hoc On-Demand Distance Vector
AP	Access Point
BFBIGM	Bird-Flocking Behaviour Inspired Group Mobility
BRP	Bordercast Resolution Protocol
BSSID	Basic Service Set Identification
C-HELLO	Cluster HELLO
C-OLSR	Cluster-based OLSR
C-RREQ	Cluster-RREQ
C-TC	Cluster Topology Control
CBRP	Cluster Based Routing Protocol
CBTRP	Cluster Based Trust-aware Routing Protocol
CCS	Connectivity-based Clustering Scheme
CEC	Clustering for Energy Conservation
CGSR	Clusterhead Gateway Switching Protocol
CID	Cluster-ID
CIDNET	Clustering for Indoor and Dense Networks
CLACR	Core Location-Aided Cluster-based Routing Protocol

ACRONYMS

ClusM	Clustered Mobility
COL	Column Mobility
CoM	Composite Mobility
Cross-CBRP	Cross Layer Cluster Based Routing Protocol
D2D	Device-to-device
DA	Disaster Areas
DefeR	Deferred Routing
DiLoC	Adaptive and Location-aware Clustering
DSDV	Destination Sequenced Distance Vector Routing Protocol
DS	Dominating Set
DSR	Dynamic Source Routing
DTN	Delay Tolerant Network
DWCA	Distributed Weighted Clustering Algorithm
ECBRP	Efficient Cluster Based Routing Protocol
ECS	Efficient Clustering Scheme
EEMC	Energy Efficient Mobility-sensitive Clustering
EMI	Electromagnetic Interference
EWCA	Entropy-based Weighted Clustering Algorithm
EWDCA	Efficient Weighted Distributed Clustering Algorithm
GPS	Global Positioning System
GSM	Global System for Mobile Communications
HBDA	Human Behaviour for Disaster Areas
HCC	Highest Connectivity Clustering
IARP	Intra-zone Routing Protocol

ID	Identification
IDs	Identifications
IERP	Inter-zone Routing Protocol
IR	Infrared
K-CONID	K-hop Connectivity ID
KhCP	K-hop Clustering Protocol
LA	Learning Automata
LACA	Learning Automata Clustering Algorithm
LBC	Load Balancing Clusters
LCC	Least Cluster Change
LCA	Lowest ID Clustering Algorithm
LSP	Link State Packet
LTE	Long-Term Evolution
LTE-Advanced	Long-Term Evolution Advanced
MAC	Media Access Control
MANET	Mobile Ad-hoc Network
Max-Min	Max-Min d-Cluster Algorithm
MEACA	Mobility and Energy-aware Clustering Algorithm
MobDHop	Mobility-based D-Hop
MOBIC	Mobility Based Metric for Clustering
MPR	Multipoint Relay
NCM	Nomadic Community Mobility
NSLOC	A Novel Stable and Low-maintenance Clustering Scheme
NSbMP	Neighborhood Stability-based Mobility Prediction

OCRP	On-Demand Clustering Routing Protocol
OHBDA	Obstacle-aware Human Behaviour in Disaster Areas
OLSR	Optimized Link State Routing Protocol
ORC	On-Demand Routing-based Clustering
PaCDS	Power-aware Connected Dominating Set
PM	Pursue Mobility
ProWBCA	Stable and Flexible Weight based Clustering Algorithm
RA	Role Advertisement
RC	Role Claim
RERR	Route Error
RFID	Radio-frequency Identification
RPGM	Reference Point Group Mobility
RREP	Route Reply
RREQ	Route Request
RSSI	Received Signal Strength Indication
RWAP	Random Waypoint with Attraction Points
RWP	Random Waypoint
SALSA	Smart and Balanced Clustering
SANE	Social Aware Networking
SCA	Stable Clustering Algorithm
SFV	Search for Victim
SMCP	Stability-Based Multi-Hop Clustering Protocol
SoCS	Social-aware Clustering Scheme
StrM	Stream Mobility

TC	Topology Control
TEDMC	Trust-related and Energy-concerned Distributed MANET Clustering
TTL	Time to Live
VC	Vote-based Clustering
VID	Virtual Identification (ID)
WCA	Weighted Clustering Algorithm
WLAN	Wireless Local Area Network
ZHLS	Zone-based Hierarchical Link State
ZRP	Zone Routing Protocol

Chapter 1

Introduction

This thesis is focused on the clustering mechanisms for wireless ad hoc networks. In this chapter, the motivating issues that led to this research work are presented in Section 1.1. Section 1.2 describes the objectives and contributions that originated from this work and Section 1.3 shows an overview of the thesis structure.

1.1 Motivation

With the advance of wireless communication technologies, small size and high performance mobile devices have been increasingly used in daily life. In a near future, a single person is expected to carry dozens of wireless devices, such as personal computers, cell phones, wearables, sensor modules and a multitude of devices capable of enhancing life experience. Meanwhile, networking technologies are gradually shifting from traditional centralised toward future distributed and self-organised, aiming at meeting the demands of a high density of devices.

Existing network infrastructures struggle to support a large amount of heterogeneous devices. A decentralised paradigm of wireless network thus becomes necessary, enabling devices to inter-connect among themselves in a distributed fashion. Here, wireless ad hoc networks play an important role, providing a self-organising network topology not requiring any kind of infrastructure to operate. In cellular communication technologies such as Long-Term Evolution Advanced (LTE-Advanced) and 5^{th} Generation (5G) mobile networks, the paradigm of Device-to-device (D2D) communication has been gaining a lot of attention in recent works [Lei et al., 2012] [Shen, 2015] [Fowler et al., 2014], aiming to accommodate the challenges inherent to a high number of devices. In the Wireless Local Area Network (WLAN) technology, the MANETs have been a popular research topic over the last two decades, mainly focusing on mechanisms to bring scalability to such networks.

MANETs are autonomous systems, capable of self-organising and self-deployment, not requiring any kind of infrastructure for their operation. The topology of such networks is highly dynamic due to unpredictable behaviour and mobility of nodes. Despite these challenges, MANETs provide many advantages. Considering their high flexibility and impromptu deployment capabilities, they can be applied to a wide range of scenarios and distinct purposes, not only where no infrastructures are available. For instance, they can be applicable to military operations, event conventions, conferences, remote areas and hostile environments, such as natural and urban disasters. Concerning the vast applications and above mentioned problems of MANETs, the development of feasible protocols and algorithms to manage such networks is a complex task. The frequent topology changes, the limited resources of nodes and radio interferences are intrinsic characteristics that must be considered when developing such protocols.

Many routing protocols were developed and evolved to face the challenging requirements of MANETs. However, these routing protocols still revealed to be insufficient when dealing with a large quantity of nodes [Xu et al., 2010]. As a result, a different approach was considered, namely the division of the network into multiple smaller groups, providing an hierarchical topology of the network. This division process is called clustering. In contrast with an entire flat topology, the clustering approach divides the network into smaller groups to improve the performance of routing protocols and hence, the scalability of the network.

Numerous clustering schemes were proposed following distinct approaches and motivated by different objectives. Whether the main focus is stability, low maintenance overhead or energy efficiency, each scheme aims at obtaining the best scalability by varying the characteristics of the algorithm, such as the election of clusterheads, the hop distance between nodes, cluster capacity and energy related policies. Although clustering enables a better performance in MANETs, a high mobility of nodes presents a great challenge, hence the large amount of research conducted around this issue. Motivated by this problem, this work studies mechanisms based on context information that will allow to increase the performance of cluster management in order to improve the network scalability.

1.2 Objectives and Contributions

The main goal of this work addresses the development of a clustering solution that is able to improve the scalability of large-scale MANETs. The proposed solution must ensure an efficient management of the cluster structure resorting to aid mechanisms so that the network is able to scale. The specific objectives are defined as follows.

- Provide an efficient clustering solution capable of handling large-scale MANETs, developing novel mechanisms to enhance network scalability

- Use context information to aid cluster management operations, while providing a stable and low overhead clustering solution

During the development of this thesis several contributions have been proposed. A brief description of the proposed solutions is provided as follows.

Smart and Balanced Clustering for MANETs

This contribution presents an alternative paradigm of clustering for MANETs. While most clustering proposals rely on centralised nodes to manage the cluster structure, SALSA is a distributed and self-organising clustering scheme assigning equal cluster management tasks to all nodes. In addition, a cluster balancing mechanism is introduced allowing nodes to be evenly distributed among clusters. Before the maximum capacity of a cluster is reached, it progressively starts assigning nodes to neighbour clusters. This contribution also proposes a cluster quality metric in order to assign nodes to the most suitable clusters, according to connectivity and free positions within clusters. The specification of SALSA resulted in the article “Smart and Balanced Clustering for MANETs”, published in the proceedings of “The 10th International Conference on Ad Hoc Networks and Wireless 2011”.

Clustering for Indoor and Dense MANETs

Motivated by the specific objective of providing scalability in dense indoor networks, CIDNET uses existent WLAN infrastructures as a mechanism of providing location references in order to expedite the cluster creation process, while maintaining well defined and evenly distributed clusters. From this solution an article named “Clustering for Indoor and Dense MANETs” was published in the proceedings of “The 15th International Conference on Next Generation Wired/Wireless Advanced Networks and Systems”.

Onto Scalable Wireless Ad Hoc Networks: Adaptive and Location-aware Clustering

In the CIDNET contribution, the clustering scheme depended on the existence of network infrastructures in order to provide location context. Motivated by this limitation, in this contribution a more versatile clustering scheme was developed, namely Adaptive and Location-aware Clustering (DiLoC). Three different approaches of clustering mechanisms were proposed. The first assumes that Wireless Local Area Network (WLAN) infrastructures are non-existent, hence it deploys several nodes with a special role, named anchor nodes, which are responsible to provide a location reference during network deployment. The second approach assumes that WLAN infrastructures exist and uses WLAN Access Points (APs) as a way of providing location references. The third approach is a combination of the previous two, acknowledging the possibility of partially existent network infrastructures. The development of DiLoC resulted in the journal article “Onto Scalable Wireless Ad Hoc Networks: Adaptive and Location-aware Clustering”, published in the Elsevier’s Journal “Ad Hoc Networks”.

Modelling Mobility based on Human Behaviour in Disaster Areas

This contribution emerges from the need to assess the performance of MANETs in a post-disaster environment. Hence, a new mobility model, named Human Behaviour for Disaster Areas (HBDA), was proposed. This mobility model attempts to mimic human movements during Search for Victim (SFV) operations in a post-disaster area. Nodes move from an initial position towards a target position, while forcing a minimum separation distance between nodes. The model is based on force vectors applied to nodes at regular time periods, until the target position is reached. From this contribution the article “Modelling Mobility based on Human Behaviour in Disaster Areas” was published in the proceedings of “The 11th International Conference on Wired/Wireless Internet Communications”.

Modelling Mobility based on Obstacle-aware Human Behaviour in Disaster Areas

This work proposed Obstacle-aware Human Behaviour in Disaster Areas (OHBDA), an extension of the HBDA mobility model which contemplates the existence of obstacles. In addition to the previous model, this contribution proposes a obstacle avoidance mechanism capable of working around obstacles of any shape. This contribution resulted in a journal article entitled “Modelling Mobility based on Obstacle-aware Human Behaviour in Disaster Areas” published in Springers Journal “Wireless Personal Communications”.

Social-aware Clustering for Wireless Ad hoc Networks

This contribution proposes a clustering scheme based on social behaviour. By providing nodes with knowledge of social patterns, SoCS is able to improve clustering performance. Each node maintains a connection history with other nodes in order to improve future cluster management decisions. From this work, the article “Social-aware Clustering for Wireless Ad hoc Networks” was published in the proceedings of the “8th Wireless Days International Conference”.

1.3 Thesis Structure

This thesis is organised and structured as follows.

Chapter 2 performs an in-depth analysis of the state of the art in clustering for MANETs. A brief description of the most relevant flat and cluster-based routing protocols for MANETs is also presented.

Chapter 3 presents Smart and Balanced Clustering (SALSA), a clustering scheme designed for multi-purpose scenarios. Moreover, a performance evaluation of clustering is also conducted.

Chapter 4 addresses the scalability issue in indoor and dense MANETs. To overcome this problem, context location information is used to assist cluster management operations. A comprehensive study of the most relevant location sensing solutions and technologies is presented. Following this study, two location aware clustering solutions are proposed to tackle the scalability issues in indoor and dense networks. Additionally, the performance of clustering schemes is evaluated by comparison.

Chapter 5 introduces a social-aware paradigm to traditional clustering in MANETs. Motivated by the information of social relations employed in routing protocols to delay tolerant networks, the Social-aware Clustering Scheme (SoCS) clustering scheme for MANETs is presented. SoCS is capable of improving clustering management based on connectivity history. SoCS is evaluated using a social mobility model, featuring both clustering and routing performance.

Chapter 6 presents a new mobility model driven to assess the network performance in post-disaster areas. The proposed mobility model attempts to mimic Search for Victim (SFV) operations in post-disaster areas, regarding the existence of diverse shaped obstacles.

Chapter 7 presents the final remarks as well as future requirements to improve this work.

Chapter 2

Solutions for the Scalability of Mobile Ad-hoc Networks

This chapter presents the State of the Art of Clustering schemes in Wireless Ad Hoc Networks. Clustering schemes are extremely important in the evolution of such networks, particularly with a large number of devices. Instead of a flat topology, a clustering scheme is able to provide a hierarchical structured network, enabling the scalability of routing protocols. Because clustering and routing are strongly tied in the development of Mobile Ad-hoc Networks (MANETs), a brief study of the most significant routing protocols is also presented.

2.1 Mobile Ad-hoc Networks

A Mobile Ad-hoc Network consists of a collection of mobile wireless nodes that self-configure to form a spontaneous network without requiring established infrastructures or any centralised management. Such networks are typically characterised by dynamic topologies, energy limitations, variable bandwidth links and fairly prone to security threats. MANETs bear a wide application potential in this type of scenarios, including disaster and emergency assistance, mobile conferences and conventions, military communications, and any kind of application that requires an impromptu fast deployed communication platform.

Self-configuration is the key feature of MANETs. There are no fixed entities that implement predefined control functions over the network. Each node is a peer to other nodes and together they organise the network cooperatively. The communication between non-neighbour nodes is relayed through intermediate nodes, serving as routers. Due to node mobility and energy constraints, the network topology is often changing, affecting the stability of the network. For example, nodes participating in a route may quit before the end-to-end communication is completed, resulting in path failure and path re-discovery [Zadin and

Fevens, 2013]. Path re-discovery bears extra delay and overhead and therefore consumes more bandwidth, significantly increasing as the number of nodes increases.

Dynamic routing is the primary concern in MANETs. It has been proven that a flat structure, either with proactive or reactive routing, does not perform well for large MANETs [Gupta and Kumar, 2000; Gerla, 2002]. With the increase of network size, a flat structured network faces scalability problems, especially with high node mobility. Consequently, a hierarchical topology is essential to guarantee a basic performance in large MANETs.

2.2 Clustering Concepts and Classification

Definition Clustering is the process of dividing the network into interconnected substructures, named clusters.

In a clustered network, nodes are divided into distinct logic groups (clusters), which are allocated geographically adjacent to each other. A typical cluster structure is shown in Figure 2.1.

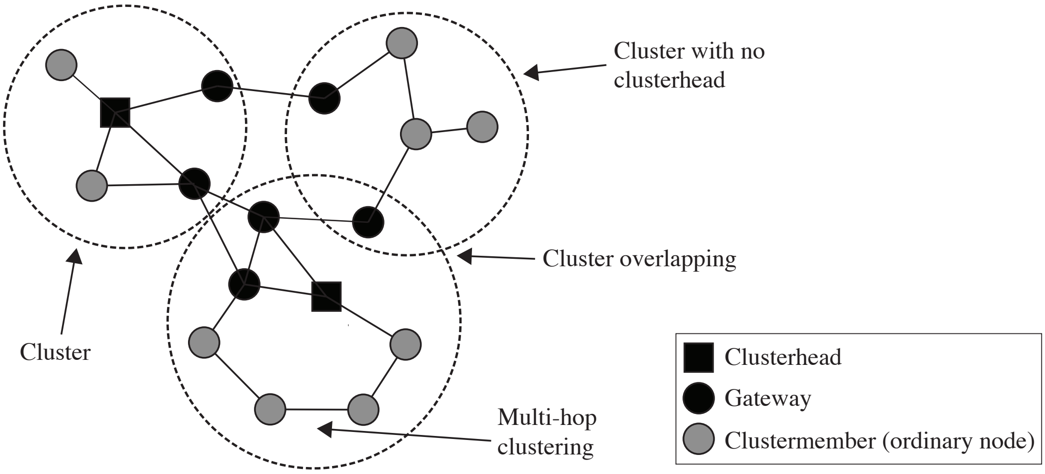


Figure 2.1: Example of a cluster structure

As depicted, nodes are divided into logic groups (within the dotted lines) according to the rules of the clustering scheme. Nodes may be assigned a different role or function, such as clusterhead, gateway or clustermember. A clusterhead typically serves as a coordinator for its cluster, performing intra-cluster management functions and data-forwarding. A gateway is a node with inter-cluster links, which can forward information between clusters. There are clustering schemes that support cluster overlapping. In other words, two distinct clusters may share nodes. In this case, gateway nodes may be assigned to more than one cluster. Finally, clustermembers, also referred as ordinary nodes, do not possess special cluster maintenance functions, they simply belong to a cluster.

2.2.0.1 Advantages of Clustering

A clustered topology in large MANETs enables efficient performance. The cluster structure provides several benefits, some of which mentioned bellow.

Reduced Topology Information Due to the number of nodes inside of a cluster being lower than the number of nodes of the entire network, the clustering process eases the aggregation of topology information. Therefore, each node is only required to store a reduced portion of the entire network routing information.

Routing Efficiency In a flat architecture every node bears equal responsibility to act as a router for forwarding packets. The great amount of message flooding inherent to path discovery reduces the routing efficiency. A clustered structure improves routing efficiency and makes the path discovery easier.

Efficiency and Stability In the perspective of a mobile node, the network appears smaller. Thus, when a mobile node disconnects or switches to another cluster, only the nodes residing in the corresponding clusters are required to modify their data structures.

There are further advantages that are transversal to the mentioned benefits. As a product of clustering, the communication bandwidth, energy consumption, throughput and scalability are improved.

2.2.0.2 Classification of Clustering Schemes

Clustering schemes can be classified according to different criteria. For example, depending on whether special nodes are required, such as clusterheads, clustering schemes can be classified as clusterhead-based and non-clusterhead-based. Or, taking into account the distance between node pairs within a cluster, schemes may be divided into 1-hop clustering and multi-hop clustering.

Given the broad types and purposes of existing clustering solutions, this study performs a classification according to their main objectives. Following this criterion, the analysed clustering schemes for MANETs are grouped into five categories, as described in Table 2.1.

In **ID-Neighbour** based clustering schemes, a unique ID is assigned to each node, and therefore each node in the network knowns the ID of its neighbours. The clusterhead is selected based on criteria involving these IDs, such as the lowest ID or highest ID. The **Connectivity-based** clustering considers the network topology around nodes. The degree of node connectivity is usually a criterion from which clustering decisions are made. The **Mobility-aware** clustering takes the mobility behaviour of mobile nodes into consideration. The mobility of nodes is the main cause of changes in the network topology. Thus, by grouping nodes with similar movement speed and pattern into the same cluster, the intra-cluster connectivity can be better stabilised. **Energy-efficient** clustering is particularly focused in the energy of mobile nodes. By eliminating or balancing the energy consumption among different nodes, the network lifetime can be prolonged. **Combined-weight** clustering usually combines multiple parameters of the previous categories. A key advantage of this category is the configuration of the individual weight factors, which can be adaptively adjusted in response to different scenarios.

Table 2.1: Classification of Clustering Schemes

Classification	**Objectives**
ID-Neighbour based	Create clusters, managed by a clusterhead elected according to the criterion of its ID.
Connectivity based	Consider the connectivity degree of nodes in order to elect the most suitable leader within clusters.
Mobility-aware	Interpret the behaviour of node mobility for cluster creation and maintenance. Assign nodes with similar relative speed to the same cluster to provide stability.
Energy-efficient	Avoid or balance energy consumption of mobile nodes in order to extend the lifetime of the network
Combined-weight	Consider multiple metrics in cluster configuration, including node degree, mobility, energy, cluster size. Adjust weight factors for different application scenarios.

2.3 Clustering in Mobile Ad-hoc Networks

As of now, a wide range of clustering schemes have been proposed for Wireless Ad hoc Networks, driven to meet different scenarios and requirements. A great amount of the existent schemes are derivations or extended versions of well known key clustering schemes, usually attempting to improve the performance in specific scenarios. This study presents the most significant clustering schemes as well as important extended contributions.

2.3.1 ID-Neighbour Based Clustering

A unique ID is assigned to each node present in the network. Typically, all nodes communicate with their neighbours using beacons (short sized broadcast messages) piggybacking their ID.

2.3.1.1 Lowest ID Clustering Algorithm (LCA)

The Lowest ID Clustering Algorithm (LCA) [Ephremides et al., 1987] is a scheme in which nodes with the minimum ID are chosen as clusterheads. A node is a Gateway if it can communicate with two or more clusterheads. The rest of the nodes are considered ordinary nodes. This is considered the most simplistic clustering method, since IDs are assigned arbitrarily disregarding any node qualifications or attributes.

2.3.1.2 Least Cluster Change (LCC)

The Least Cluster Change (LCC) [Chiang et al., 1998] algorithm is an improvement of the Lowest ID Clustering Algorithm (LCA) which attempts to minimise the cost of re-clustering operations. When an ordinary node moves outside its cluster and does not join another cluster then it becomes a clusterhead forming a new cluster. The LCC algorithm is divided in two stages, the cluster formation and cluster maintenance. The formation simply follows the LCA, where nodes with the lowest ID are elected as clusterheads. The maintenance stage is event-driven and can be triggered by the two following events. When a clusterhead becomes at 1-hop distance from any clusterhead, the one with lowest ID gives up its clusterhead role. When an ordinary node moves outside its cluster and does not join another cluster then it becomes a clusterhead, creating a new cluster. LCC is able to significantly improve stability at short term, when compared to LCA. However, a single node movement may trigger a complete cluster structure re-calculation leading to a large communication overhead.

2.3.1.3 Max-Min d-Cluster Algorithm (Max-Min)

The Max-Min d-Cluster Algorithm (Max-Min) [Amis et al., 2000] forms clusters with nodes that are distanced at most d-hops from a clusterhead. The value of d controls the number of elected clusterheads and consequentially the density of clusterheads in the network. Furthermore, each node is limited to send a multiple of d number of control messages. The election of clusterheads is based on a two round flooding. After this flooding process, the nodes with the smallest node ID are elected as clusterheads.

2.3.1.4 Summary of ID-Neighbour Based Clustering

The ID-Neighbour based clustering schemes have the ability to create the cluster structure in a short period of time, since the only criterion to elect the clusterheads is the ID of nodes. However, this benefit is sacrificed in terms of inherent bias towards nodes with smaller IDs. These nodes typically suffer from quick battery drainage without the attempt to balance

Table 2.2: Summary of ID-Neighbour Based Clustering

Scheme	**CH**	**Hops**	**OL**	**Extends**	**Objectives**
LCA	Yes	1-Hop	No	–	Low complexity
LCC	Yes	1-Hop	No	LCA	Reduce overhead by eliminating unnecessary maintenance operations
Max-Min	Yes	n-Hops	No	LCA	Reduce overhead by limiting the amount of control messages

CH: Clusterhead-based clustering
Hops: Maximum number of hops inside clusters. If clusterhead-based, between any node and the clusterhead. Otherwise, between any two nodes
OL: Overlapping clusters

the load across other nodes. Table 2.2 shows the characteristics for the presented clustering schemes, regarding their objectives.

2.3.2 Connectivity-based Clustering

Connectivity-based clustering schemes determine the connectivity degree, i.e. the amount of connection links, between a node and its direct neighbours. Each node broadcasts periodically its connectivity value to its direct neighbours. Typically, the node with the highest connectivity degree in the neighbourhood is assigned the clusterhead role.

2.3.2.1 Highest Connectivity Clustering (HCC)

The Highest Connectivity Clustering (HCC) scheme [Gerla and Tzu-Chieh Tsai, 1995] forms clusters according to the connection degree. The degree of each node is calculated based on the hop distance from its neighbours. Nodes broadcast their ID to neighbours. The node with the highest amount of neighbours (highest degree) is elected as clusterhead. The maximum distance between any two nodes in the cluster is 2-hops, since the clusterhead is at 1-hop distance from all member nodes. HCC reduces the number of clusterhead re-elections, however, as the number of node members increases, the clusterhead throughput decreases. Furthermore, the HCC algorithm does not set a limit on the amount of nodes per cluster, which will typically result in overloaded clusterheads.

2.3.2.2 K-hop Connectivity ID (K-CONID)

The K-hop Connectivity ID (K-CONID) clustering scheme [Nocetti et al., 2002] elects clusterheads based on two criteria. The connectivity node degree is considered as first criterion and lowest node ID as second criterion. The combination of the two criteria mitigates the problem of a single node with a very high connectivity two overcome others. On the other hand, the lowest ID criterion generates a large amount of clusters. K-CONID attempts to minimise the number of clusters by combining these two election criteria. Clusters are formed with nodes that distance at most k-hops from the clusterhead, begin k an configurable arbitrary parameter.

2.3.2.3 3-hop Between Adjacent Clusterheads (3-hBAC)

The 3-hop Between Adjacent Clusterheads (3-hBAC) scheme [Yu and Chong, 2003] introduced a new node status, the clusterguest. A node in this status is not in transmission range of any clusterhead, but it can be connected to some clustermembers. The formation of clusters always begins within the neighbourhood of the node with the lowest ID in the network. In this neighbourhood, the node with the highest degree is elected as clusterhead. All the direct neighbours of this clusterhead are assigned as clustermembers. Furthermore, the neighbours of each clustermember, which have their status unassigned, are denied the role of clusterhead, forcing a distance of at least 3 hops between clusterheads. Any node to which has not been denied the role of clusterhead declares itself as such once it has the highest degree in its neighbourhood.

2.3.2.4 Stable and Flexible Weight based Clustering Algorithm (ProWBCA)

The Stable and Flexible Weight based Clustering Algorithm (ProWBCA) [I Selvam, 2011] is a clustering scheme based on the connectivity of nodes. Two main types of nodes exist, namely normal nodes and multicast nodes. Their role is pre-configured before the network start up. The election of the clusterhead is based on the weight formula $w(p) = 3t(p) + 2s(p) + r(p)$, where $t(p)$ and $s(p)$ are the number of multicast neighbour nodes at 1-hop and 2-hops, respectively. $r(p)$ represents the sum of the multicast and normal nodes within 2-hops. After the weight calculation, the node which has the largest weight will declare itself as clusterhead. This metric elects the node with highest connectivity in the neighbourhood to be a clusterhead, while ensuring that the maximum distance between any clustermember and the clusterhead is 2-hops.

2.3.2.5 Connectivity-based Clustering Scheme (CCS)

The Connectivity-based Clustering Scheme (CCS) [Mai and Choo, 2008] is a multi-hop scheme with the purpose of improving the effectiveness, reliability and stability of MANETs. CCS is based on the assumption that a high connectivity degree as metric of electing clusterheads alone is not sufficiently reliable to maintain cluster stability. Instead, since clusters are likely to become more stable when their size is large, CCS is based on multi-hop clustering. The k parameter is a configurable value which dictates the number of hops allowed within each cluster. Moreover, based of the K parameter, the transmission power is adjusted accordingly.

CCS also provides balancing and removal of unnecessary clusters. To reduce the number of small clusters and improve effectiveness, several conditions are applied. When a node determines that its cluster size is small, its distance to the clusterheads is larger than K or its connection degree is below a threshold, it considers to join another cluster. This mechanism provides a method of progressive cluster balance, avoiding large re-clustering operations.

2.3.2.6 Summary of Connectivity-based Clustering

Typically, a high connection degree allows clusterheads to have more clustermembers that are in their neighbourhood and the cluster size may be larger. The summary of the studied schemes that rely on this metric is presented in Table 2.3. The HCC scheme is the most simple connectivity-based scheme. It directly assigns the clusterhead role to nodes with highest degree in a one hop distance. The K-CONID and 3-hBAC schemes extend and combine the HCC and LCA clusterhead election metrics.

Table 2.3: Summary of Connectivity-based Clustering

Scheme	**CH**	**Hops**	**OL**	**Extends**	**Objectives**
HCC	Yes	1-Hops	No	-	Provide cluster stability by electing clusterheads with best connectivity
K-CONID	Yes	K-Hops	No	LCA HCC	Combines the lowest ID and highest degree heuristics balance the number of clusters
3-hBAC	Yes	1-Hops or 2-Hops, if clusterguest	No	LCA HCC	Extend cluster connectivity by allowing clusterguests. Force non-overlapping clusters with a 3-hop distance between clusterheads
ProWBCA	Yes	2-Hops	No	-	Stability increase by assuring the centrality and connectivity of the clusterhead
CCS	Yes	n-Hops	No	-	Effectiveness, reliability and stability with progressive cluster balancing

CH: Clusterhead-based clustering
Hops: Maximum number of hops inside clusters. If clusterhead-based, between any node and the clusterhead. Otherwise, between any two nodes
OL: Overlapping clusters

2.3.3 Mobility-aware Clustering

Mobility is one of the most relevant attributes of MANETs, since it is the movement dynamics that affects the cluster topology and causes route invalidation. Understanding the mobility behaviour of mobile nodes to manage the cluster structure is the concept behind mobility-aware clustering schemes. Intra-cluster links can become more reliable if nodes with similar movement behaviour are grouped into the same cluster. This will naturally decrease the re-clustering rates.

2.3.3.1 Mobility Based Metric for Clustering (MOBIC)

The Mobility Based Metric for Clustering (MOBIC) [Basu et al., 2001] enforces that clusterhead election should take mobility of nodes into consideration. In MOBIC, the distance between nodes is calculated based on received signal strength. With this distance, being periodically measured at distinct time instants, it is determined the relative mobility of nodes. Nodes with lowest relative mobility are chosen as clusterheads as they will likely provide more stability. The cluster maintenance of MOBIC is similar to LCC [Chiang et al., 1998] with an additional mechanism to decrease the overhead. The Cluster Contention Interval (CCI) is introduced to avoid unnecessary clusterhead re-election when two clusterheads become in range for a short period of time. If two clusterheads become in range of each other for a time period larger than the CCI value, one gives up the clusterhead role.

MOBIC is designed to handle group mobility behaviours within clusters, whereas a group of nodes move with similar speed and direction. Within this group, the node with the lowest speed relative to its neighbours is the clusterhead. However, if nodes move randomly, changing their direction and speed, the performance of MOBIC will be highly affected.

2.3.3.2 Mobility-based D-Hop (MobDHop)

The Mobility-based D-Hop (MobDHop) clustering scheme [Er and Seah, 2004] forms variable diameter clusters based on the MOBIC mobility metric [Basu et al., 2001]. Clustermember nodes can be at most d-hops from the clusterhead. The d value is adaptable with respect of node mobility characteristics. MobDHop assumes each node is able to measure received signal strengths in order to determine distances to its neighbours. Distances from neighbours are calculated in two time instants in order to determine if two nodes are moving away from each other or if they are becoming closer. As a result, the d value is dynamically adjusted allowing more or less hops between clustermember nodes and the clusterhead. A local stability metric is defined as being the variation of distances between neighbours. The node with the lowest value of local stability is elected as clusterhead.

This algorithm tolerates high node mobility before triggering re-clustering operations, therefore reducing overhead. However, MobDHop has a significant limitation. The measurement of physical distance between nodes is not precise. For instance, a node may transmit packets at a low power, thus being perceived as a distant node when, in fact, it is physically close.

2.3.3.3 Mobility and Energy-aware Clustering Algorithm (MEACA)

The Mobility and Energy-aware Clustering Algorithm (MEACA) [Xu and Wang, 2006] measures de node mobility and energy parameters to elect the most stable clusterhead. Similarly to MOBIC, it also avoids premature clusterhead re-election to increase cluster stability. Nodes in the network have different priorities to become clusterhead. Their priority values are exchanged to determine which nodes will become clusterheads. The priority values are deterministic, allowing nodes in the network to reach unanimous decisions on their roles.

Unlike the LCA algorithm, that predefines the priority level of a node to become clusterhead, MEACA sets the priority using the mobility and energy status of nodes. Each node has a *mobility attribute* and a *energy attribute*, which change dynamically. Within each neighbourhood, the clusterhead will be the node with the relatively lowest mobility and highest energy. Re-clustering operations are kept to a minimum. They only occur when a clustermember has lost contact with its clusterhead or the clusterhead has lost contact with all of its clustermembers.

2.3.3.4 Neighborhood Stability-based Mobility Prediction (NSbMP)

In order to reduce the number of re-clustering operations caused by node mobility, the Neighborhood Stability-based Mobility Prediction (NSbMP) clustering algorithm [Konstantopoulos et al., 2008] was proposed. The NSbMP scheme attempts to predict the future mobility of nodes, assigning a weight to each node according to its mobility probability. Nodes with the highest weights (i.e. lower mobility probability) are chosen as clusterheads. Following this procedure, the NSbMP builds a virtual backbone covering the entire network, which is highly resistant to node mobility.

2.3.3.5 Learning Automata Clustering Algorithm (LACA)

The Learning Automata Clustering Algorithm (LACA) [Xun, 2013] relies on the theory of Learning Automata (LA) to manage cluster structure. LA is an adaptive decision-making system, usually utilised in unknown environments aiming at reaching the best decision with provided data. In other words, the decisions taken in LACA are based on a probability distribution kept and updated as the network topology changes. Each node executes a LA algorithm which is fed with network parameters through the exchange of `HELLO` messages.

Each node periodically broadcasts `HELLO` messages indicating their availability, location and velocity. With such parameters, two metrics are derived, namely the distance between neighbour node pairs, and the future distance between every two neighbour nodes. Clusterheads are elected regarding this prediction, with the expectancy that nodes remain stable in the same cluster. LACA forms a cluster structure with a one hop neighbourhood.

2.3.3.6 Summary of Mobility-aware Clustering Schemes

Mobility phenomena is the main factor causing instability to the cluster structure. Mobility-aware schemes attempt to mitigate this problem by acknowledging the mobility of nodes during management decisions.

Table 2.4: Summary of Mobility-aware Clustering Schemes

Scheme	CH	Hops	OL	Extends	Objectives
MOBIC	Yes	1-Hop	Yes	LCC	Minimize re-clustering operations caused by node mobility within clusters
MobDHop	Yes	n-Hops	No	MOBIC	Support of wide area clusters; minimise number of clusterheads
MEACA	Yes	1-Hop	No	-	Create clusters considering the mobility and energy of nodes
NSbMP	Yes	n-Hops	No	-	Minimise mobility impact using node mobility prediction.
LACA	Yes	1-Hop	No	-	Continuously predict relative node mobility to form high stability clusters

CH: Clusterhead-based clustering
Hops: Maximum number of hops inside clusters. If clusterhead-based, between any node and the clusterhead. Otherwise, between any two nodes
OL: Overlapping clusters

Table 2.4 shows the main characteristics of the presented clustering schemes. While some schemes merely attempt do reduce the impact of mobility by the employment of restrictions (MOBIC and MobDHop), others consider and predict node mobility to improve future decisions (MEACA, NSbMP and LACA).

2.3.4 Energy-efficient Clustering

Mobile nodes in a MANET depend on battery power supply during operation, which pose challenges regarding energy limitation for network performance. A clustering scheme should be designed to save this resource as far as possible in order to prolong the network lifetime. The presented clustering schemes take as primary consideration the energy efficiency while maintaining the cluster structure.

2.3.4.1 Load Balancing Clusters (LBC)

The Load Balancing Clusters (LBC) scheme [Amis and Prakash, 2000] defines a Virtual ID (VID) to each node, initially assigned to its ID number. During network startup, LBC acts as the LCA scheme, assigning the clusterhead role to nodes with the highest ID. However, LBC limits the maximum time units that a node can serve as clusterhead. When a clusterhead depletes its role duration time it resets its VID to 0, thus becoming an ordinary node. Each ordinary node keeps a circular queue for its VID, shifting its value by one per time unit. As a result, when a clusterhead resigns, the ordinary node with the highest VID in the neighbourhood becomes the next clusterhead. Furthermore, every time two clusterheads become in communication range, the one with the lowest VID gives up its role, becoming an ordinary node.

With this methodology, LBC tries to avoid possible node failures due to energy depletion by distributing the load of clusterhead tasks across multiple nodes. However, the re-clustering operations cased by periodic clusterhead re-election may introduce unnecessary overhead. In addition, the clusterhead serve timer alone may not realistically represent the amount of energy consumption of a node.

2.3.4.2 Power-aware Connected Dominating Set (PaCDS)

The Power-aware Connected Dominating Set (PaCDS) [Stojmenovic, 2001] is an energy-efficient scheme which is able to increase or decrease the size of the Dominating Set (DS) according to the energy level of nodes. Nodes inside the DS have more tasks, such as packet relay and routing information updates, therefore consuming more energy than remaining nodes. PaCDS elects clusterheads based on the energy level of nodes, instead of ID or node degree as seen in other schemes. When possible, nodes with low residual energy are removed from the DS, allowing them to extend their lifetime in the network. On the other hand, when a node is removed from the DS, the load of nodes inside the DS increases and they will likely deplete energy at a faster rate.

2.3.4.3 Clustering for Energy Conservation (CEC)

In Clustering for Energy Conservation (CEC) [Ryu et al., 2002] a node can be either a master (clusterhead) or a slave (ordinary). A slave node can connect to only one master node and direct connections between slaves are not allowed. Master nodes are previously selected offline and each can only connect to a limited number of slave nodes. CEC is designed to minimize the used transmission power between each master-slave pair in order to extend network lifetime.

Two steps of operation are proposed in CEC, namely single-phase and double-phase clustering. In single-phase clustering, initially every master node will page slave nodes with the maximum transmission power. Upon receiving the page, each slave sends an `ACK` message to the master with the strongest paging signal. Since a master node can only serve a limited number of slaves, it first allocates channels to slaves that only received a single page signal (from itself). Then, the remaining channels (if any) are allocated to remaining slaves that sent an `ACK`. The slaves that did not receive any pages are addressed in the double-phase. This mechanism forms links with minimum distance between each master-slave pair, allowing to reduce the required transmission power in order to save energy. Each master, which has free channels after the single-phase, initiates the double-phase clustering by sending a page with a stronger transmission power in order to reach unallocated slave nodes. The rest of the procedure is equal to the single phase clustering. Despite the proposed scheme nearly achieves optimal performance [Ryu et al., 2002], the method of selecting the master nodes is not defined. In addition, the methodology of maintaining the cluster structure upon node mobility is not described.

2.3.4.4 Energy Efficient Mobility-sensitive Clustering (EEMC)

The Energy Efficient Mobility-sensitive Clustering (EEMC) [Wei et al., 2007] algorithm aims at balancing the power consumption amongst nodes and clusters of the network. The speed of nodes is monitored in order to determine the flooding rate of control messages. Furthermore, a minimum and maximum number of nodes per cluster is statically established before network initialisation.

In addition to the mentioned features, this algorithm also performs route discovery. By the utilisation of beacon messages, this protocol can form cluster structures and route discoveries in a concurrent approach. The EEMC algorithm also relies on clusterheads for cluster coordination. For intra-cluster communication a proactive routing protocol is utilised, whereas inter-cluster communication depends on a reactive routing protocol. As a result, intra-cluster updates are performed utilising beacon messages, in order to keep each cluster's topology updated. Furthermore, contrary to traditional clustering algorithms, this algorithm does not send fixed periodic beacon messages. The rate of beacon transmission is calculated according to the mobility speed of the nodes and can be adapted through time. By doing so, a large amount of control overhead is avoided, since low motion nodes require fewer update messages.

In order to balance power consumption between nodes, the EEMC algorithm distributes the role of clusterhead between the nodes within the cluster. This way, the complete energy drain of a single node is avoided, extending the network lifetime at the cost of increased complexity. Moreover, the cluster size is also taken into account, as the bigger a cluster is, the higher is the weight carried by its clusterhead. Thus, the algorithm attempts to evenly distribute the nodes amongst all clusters, resulting on nearly equal size clusters.

2.3.4.5 Trust-related and Energy-concerned Distributed MANET Clustering (TEDMC)

The Trust-related and Energy-concerned Distributed MANET Clustering (TEDMC) [Qiang et al., 2008] is also a scheme driven by energy concerns. TEDMC considers that the most important nodes are the clusterheads, and therefore it elects them according to their trust level and residual energy. In order to keep information about the trust level of nodes, this algorithm maintains and periodically exchanges a reputation rank table, which contains a reputation value and the unique identification of the last node to assign the value in question. Furthermore, TEDMC is substantially different from K-hop Clustering Protocol (KhCP), as it only allows 1-hop clusters, thus being less suitable for dense networks.

2.3.4.6 Summary of Energy-efficient Clustering

In clusterhead based clustering schemes, clusterheads are essential to several administrative tasks and inter-cluster communication. The sudden failure of such nodes is typically associated with large re-clustering operations and therefore most of the presented schemes mainly address the protection of clusterhead failure due to energy loss. The described schemes are shown in Table 2.5.

2.3.5 Combined-weight Clustering

Combined-weight clustering schemes consider a number of distinct metrics for cluster configuration. They permit a certain flexibility by adjusting the weighting factors for each metric to adapt to a variety of scenarios.

2.3.5.1 Weighted Clustering Algorithm (WCA)

The Weighted Clustering Algorithm (WCA) [Chatterjee et al., 2000] combines four distinct metrics for the clusterhead election, namely the degree-difference (D_v), the sum of distance

Table 2.5: Summary of Energy-efficient Clustering

Scheme	**CH**	**Hops**	**OL**	**Extends**	**Objectives**
LBC	Yes	1-Hop	No	–	Periodically change clusterhead to avoid failures caused by energy depletion
PaCDS	Yes	1-Hop	No	–	Exclude low energy nodes from crucial tasks, such as packet relaying and routing updates
CEC	Yes	1-Hop	No	–	Reduce transmission power based on geographical distance of nodes
EEMC	Yes	n-Hops	No	–	Limit control messages based on node mobility. Distribute power consumption between nodes
TEDMC	Yes	1-Hop	No	–	Election of clusterheads based on trust level and residual energy

CH: Clusterhead-based clustering
Hops: Maximum number of hops inside clusters. If clusterhead-based, between any node and the clusterhead. Otherwise, between any two nodes
OL: Overlapping clusters

to all neighbours (P_v) the average node speed (M_v) and the amount of time serving as clusterhead (Tv). The D_v metric is given by $D_v = |d_v - M|$, where d_v is the number of neighbours of node v and M is the maximum number of nodes that a clusterhead can support. The combined weight value, Iv, is determined by $I_v = c1D_v + c2P_v + c3M_v + c4T_v$, being $c1$, $c2$, $c3$ and $c4$ the weight factor constants that can be chosen to meet the requirements of the target scenario. The mobile nodes with minimum I_v value are chosen to be clusterheads. The clusterhead selection algorithm is invoked in two situations, at the beginning of cluster formation and whenever clusterheads are not able to cover all the nodes to which had a previous connection. WCA does not perform re-clustering operations when a node changes its cluster. This mechanism ensures maximum cluster stability, however, it does not strictly follow the minimum I_v for each node, otherwise it would have to update clusterheads every time a node changes cluster. When a node is not in communication range with any clusterhead, the clusterhead selection is invoked repeatedly until all nodes in the network are covered. This mechanism may result in a great amount of overhead. In a dynamic network, the constant clusterhead re-election may require a significant amount of message exchange, thus the WCA algorithm may not be suitable for typical mobile ad hoc networks.

2.3.5.2 Entropy-based Weighted Clustering Algorithm (EWCA)

The Entropy-based Weighted Clustering Algorithm (EWCA) [Wang and Bao, 2007] introduces a mechanism to mitigate the network overhead of WCA. The high mobility of nodes in WCA lead to a high frequency of clusterhead updates, generating high communication overhead. EWCA overcomes this drawback with an entropy-based model for the clusterhead election. In the EWCA scheme a Tabu Search (TS) approach is used to optimise the clusterhead selection algorithm, which is able to reduce the number of clusterheads and the amount of re-clustering operations. Entropy presents uncertainty and is a measure of disorder in a system, thus being better at delivering stability with a lower overhead.

2.3.5.3 Stability-Based Multi-Hop Clustering Protocol (SMCP)

Stability-Based Multi-Hop Clustering Protocol (SMCP) [Tenhunen et al., 2005] builds and maintains multi-hop clusters, using the associativity and link quality of nodes as its main criteria. The cluster structure is created and maintained using a constant exchange of control messages. In addition to this, the proposed scheme introduces a new concept, namely the *clustercast* mechanism in order to reduce forwarding of broadcast control messages.

For neighbour discovery, SMCP utilises `HELLO` messages which are transmitted within a period of time. Even after the entire cluster structure is formed, nodes keep transmitting `HELLO` messages, allowing the protocol to constantly monitor the nodes along with their connection quality and stability. The clusterhead selection algorithm is based on several metrics, with the purpose of obtaining the most suitable node, such as degree of connectivity, stability of the link to neighbours and available battery level. The selecting process utilises two types of messages, namely the Role Advertisement (RA) and the Role Claim (RC), which are exchanged amongst stable nodes.

This protocol introduces a new methodology with the purpose of restraining certain control messages. The broadcast of control messages is costly and should be minimised as much as possible. The *clustercast* mechanism enables stable clusters to be monitored more often than unstable clusters, in order to constraint unnecessary flooding in the latter.

2.3.5.4 Vote-based Clustering (VC)

The Vote-based Clustering (VC) algorithm [Kahng and Goto, 2004] is based on two metrics to elect the clusterhead, namely the number of neighbour nodes (node degree) and the amount of remaining energy, expressed as battery time. A voting method is used to select the clusterhead, described as follows. Each node (MH) broadcasts an `HELLO` message (Table 2.6) which contains its ID (`MH_ID`), its clusterhead ID (`CH_ID`) and the `VOTE`, which is the node calculated vote value.

Table 2.6: Hello message format of Vote-based clustering

`MH_ID`	`CH_ID`	`VOTE`	`OPTION`

The Vote value is defined as $Vote = w1 \times (n/N) + w2 \times (m/M)$, where $w1$ and $w2$ are the weight coefficients for node degree and amount of battery time, respectively. n represents the number of neighbours for a maximum number of nodes in a cluster N. m is the remaining battery time for a maximum battery life time of M.

Each node sends an `HELLO` message at a random time during an *Hello* cycle. If a node is a new node in the network, the `CH_ID` value is reset, not being assigned to any cluster and not being aware of any neighbour member nodes. Each node registers the value n based on the number of received `HELLO` messages. Then, it sends another `HELLO` message setting the Vote value, calculated according to the $Vote$ formula. At the end of two `HELLO` cycles every node in the network will have knowledge of its clusterhead, which is the sender node with the highest vote. VC also provides a method to limit the amount of nodes per cluster. Clusterhead nodes use the `OPTION` value of the `HELLO` message to inform the current number of their cluster members. When this number exceeds a threshold (maximum number of nodes per cluster) no more nodes will be assigned to its cluster.

2.3.5.5 Stable Clustering Algorithm (SCA)

The Stable Clustering Algorithm (SCA) [Tolba et al., 2007] aims at supporting large MANETs containing nodes moving at high speeds by reducing re-clustering operations and stabilizing the network. To meet these requirements, the algorithm is based on the quick adaptation to the changes of the network topology and reduction of clusterhead reelections. In order to avoid a high frequency of clusterheads reelection, the algorithm initially chooses the nodes that best meet some required metrics such as, energy, mobility, connectivity and communication range.

2.3.5.6 Distributed Weighted Clustering Algorithm (DWCA)

The Distributed Weighted Clustering Algorithm (DWCA) [Choi and Woo, 2006] protocol is an improvement of the Weighted Clustering Algorithm (WCA), introducing new features that nowadays are required to achieve a satisfactory performance on MANETs. The new proposed algorithm aims at a distributed clustering set up, extending the network lifetime. Therefore, the protocol has been divided in two phases of operation, the cluster set up and the cluster maintenance.

In the cluster set up phase, the algorithm constructs the cluster structures, electing clusterhead nodes to handle the cluster management. The nodes broadcast `HELLO` messages in order to discover their neighbours, maintaining a list of nearby neighbours. The election of clusterheads is based on the weight value of the nodes. Each node calculates its weight value according to metrics such as the number of nodes that the node itself can ideally handle, the sum of distance to all its neighbours, the movement speed and the remaining battery energy.

Cluster maintenance operations are performed in a passive manner. Only two situations require a clustering reorganisation, one is the node movement leaving the cluster range and the other is the low battery of a clusterhead.

2.3.5.7 Efficient Weighted Distributed Clustering Algorithm (EWDCA)

The Efficient Weighted Distributed Clustering Algorithm (EWDCA) [Hussein et al., 2010] is also an extension of the WCA scheme with the concern of providing scalability for MANETs, by taking into consideration several weight parameters: connectivity, residual battery energy, average mobility and distance between nodes. These parameters are used only to elect the most suitable clusterhead, in order to keep an optimal number of clusters, thus providing as much scalability as possible.

2.3.5.8 Summary of Combined-weight Clustering

In literature, the combined-weight clustering is probably the most explored solution aiming to meet different scenario requirements. In this analysis, some of the most relevant clustering schemes were presented and described. Their characteristics are summarised in Table 2.7.

Table 2.7: Summary of Combined-weight Clustering

Scheme	**CH**	**Hops**	**OL**	**Extends**	**Objectives**
WCA	Yes	1-Hop	No	–	Select most suitable clusterheads based on multiple metric with distinct impact factors, keeping a stable cluster topology
EWCA	Yes	1-Hop	No	WCA	Reduce WCA overhead by introducing uncertainty in clusterhead election metric
SMCP	Yes	n-Hops	No	–	Provide stability with link quality monitoring
VC	Yes	1-Hop	No	–	Create and maintain clusters based on node degree and remaining battery time, according to a vote-based system. Restrict maximum number of nodes per cluster

SCA	Yes	1-Hop	No	–	Provide stability based on four metrics: energy, mobility, connectivity and communication range
DWCA		1-Hop	No	WCA	Extend network lifetime based on node degree, mobility and energy
EWDCA	Yes	1-Hop	No	WCA	Increase scalability by considering connectivity, residual energy, mobility and geographical distance between nodes

CH: Clusterhead-based clustering
Hops: Maximum number of hops inside clusters. If clusterhead-based, between any node and the clusterhead. Otherwise, between any two nodes
OL: Overlapping clusters

The WCA was one of the pioneers in this category, which served as a pillar of many other clustering schemes. Even tough SCA is not an extension of WCA, its objectives and used metrics are very similar to WCA.

2.4 Routing in Mobile Ad-hoc Networks

In recent years, routing protocols for Ad Hoc Networks have been a subject of intensive study and innovation given that traditional routing protocols were designed to operate in wired networks. The dynamic topology and metric-awareness that is required by MANETs to efficiently perform, do not allow the utilisation of a traditional routing protocol. Therefore, many efforts have been done in order to achieve an efficient routing protocol, by the modification of traditional protocols, the combination of multiple protocols or by the creation of new solutions.

Routing protocols are generally classified into reactive and proactive protocols. The reactive routing protocols perform path discovery on-demand. Proactive protocols, on the other hand, determine the routes in advance keeping all the existent routes updated on each mobile node of the network. Both types of protocols can be advantageous depending on the target application scenarios or purposes. Reactive routing protocols perform `Route Request (RREQ)` flooding for route discovery, bearing extra additional delay, but potentially avoiding unnecessary overhead in updating routes that may not be utilised. This type of protocols can be especially useful for MANETs with a large number of mobile nodes and a relatively low amount of messages exchange. Proactive protocols keep all mobile nodes with updated routes by constantly exchanging information between the mobile nodes of the network. This approach produces a considerable amount of overhead in the network, which increases significantly as the number of mobile nodes grows.

In addition to the previously mentioned protocols, an hybrid approach is also considered, which combines the advantages of proactive and reactive protocols. Most hybrid routing protocols start by finding all the routes in a proactive way, which are then updated and maintained on-demand.

2.4.1 Proactive Routing Protocols

As mentioned before, proactive routing protocols maintain the routes to all destinations regardless of the fact that the routes may not be utilised. Therefore, all the existing nodes in the network must periodically exchange control messages in order to keep routes updated. Consequently, proactive protocols could unnecessarily waste resources by updating routes that will never be utilised in the future. However, this type of protocols can quickly deliver messages which represents an important advantage.

2.4.1.1 Optimized Link State Routing Protocol (OLSR)

The Optimized Link State Routing Protocol (OLSR) [Jacquet et al., 2001; Clausen and P. Jacquet, 2003] is an evolution of the classical link state protocols, specially optimised for Wireless Ad-Hoc Networks. Being a proactive routing protocol it calculates all the possible paths in advance, constantly sending `HELLO` and `Topology Control (TC)` messages along the network in order to discover the available routes. The OLSR protocol uses Multipoint Relays (MPRs) to forward control messages and minimise control overhead.

Each node periodically emits `HELLO` messages to its neighbours, containing a list of neighbours known to the node. The `HELLO` messages reach only the node's one-hop neighbours, as they are not forwarded any further. Based on the retrieved information the node elects a group of MPRs amongst its one-hop neighbours. The selection algorithm of MPRs consists of an interactive process which discards the less suitable nodes in the process, remaining a typically small set of nodes (MPRs) at the end. The selected neighbours are the ones that have the best connectivity to the node's two-hop neighbours.

The OLSR protocol maintains, on each node, the topology of the network using `TC` messages. Only nodes selected as a MPR are allowed to transmit such messages. The `TC` messages carry the group of nodes that selected the source node as being one of its MPRs. With the utilisation of MPRs it is possible to optimise the control messages overhead, as each node only forwards messages that are received for the first time and that are originated from nodes that selected it as an MPR. Consequently, each node is only reachable by its one-hop neighbours or by its MPRs.

2.4.1.2 Destination Sequenced Distance Vector Routing Protocol (DSDV)

One of the first loop-free routing protocols developed for MANETs was the Destination Sequenced Distance Vector Routing Protocol (DSDV) [Perkins et al., 1994]. Sequence numbers are used on each entry of the routing table, keeping the most updated routing paths and avoiding routing loops.

In order to maintain consistency on a constantly changing topology, update messages are periodically exchanged between nodes, and immediately transmitted upon detection of new routing information.

In the DSDV protocol, each node periodically advertises its own routing table to its neighbours, occasionally the full table or merely some entries of it. Typically the full table is sent upon network initialisation while entries are exchanged to keep the routing tables updated. However, the full table can also be shared after protocol initiation but occurs infrequently. Each broadcast message transmitted by a mobile node contains a new sequence number with the purpose of keeping the most recent path. The receiving nodes will also advertise their routes to their neighbours, consequently maintaining the network updated.

The DSDV protocol requires the transmission of a large amount of packets in order to keep the routing paths updated. Hence, it may not be very suitable for MANETs with a large number of nodes as it consumes a significant amount of bandwidth and a great amount of energy, even when data transmissions are idle.

2.4.2 Reactive Routing Protocols

The reactive routing protocols can considerably reduce the overhead caused by the exchange of control messages since they do not constantly maintain the routes updated. Furthermore, reactive routing protocols only establish routes by nodes that specifically require them, excluding the overhead of control messages destined do update routes that may not be used in the future. However, source nodes have to wait for the routing discovery before they can send any data, resulting in high response times.

2.4.2.1 Ad-Hoc On-Demand Distance Vector (AODV)

Ad-Hoc On-Demand Distance Vector (AODV) [Perkins and Royer, 1999; Perkins et al., 2003] protocol was especially designed for MANETs. Being a reactive protocol it only discovers the routes on demand, meaning that routes are established only when the source node is required to send a message. Like most of the distance vector routing protocols, AODV is loop-free by using sequence numbers on every route update, a scheme originally developed in the DSDV routing protocol.

AODV uses `RREQ` and `Route Reply (RREP)` messages in order to build routes. If a node wishes to transmit a message for which it does not have an established route, it first broadcasts a `RREQ` message along the network. The receiving nodes update their tables using the information on the `RREQ` message, such as the source IP address, sequence number and broadcast ID. If the receiving nodes possess a more recent route (i.e. with a higher sequence number) they can reply directly to the source with a `RREP` message, otherwise they just propagate the message. Furthermore, if a node receives a duplicated `RREQ`, the message is discarded and not forwarded any further. Finally, as soon as the `RREQ` originating node receives a `RREP` message, it means that the required route is ready to be utilised and, as a result, data can be sent.

After data transmission the routes will only be active for a certain period of time, being completely removed as soon as the timer expires. However, if some connection is broken along an active routing path, the closest node to the breakout transmits a `Route Error (RERR)` message back to the source, forcing the restart the route discovery process.

2.4.2.2 Dynamic Source Routing (DSR)

Dynamic Source Routing (DSR) [Johnson et al., 2007] is an on-demand protocol specifically developed for MANETs. The DSR protocol is very similar to AODV regarding route discovery, but utilises a completely different mechanism to maintain the routing information. Contrary to AODV which maintains a routing table per node, DSR is based on source routing. That is, the originating node knows the complete routing path towards the destination.

As mentioned before, DSR makes use of the same mechanism as AODV to perform route discovery. When a node in the network requires to send data to a destination for which it does not hold the correct path, it floods the network with `RREQ` packets. As long as `RREQ` messages are traveling in the network, they are building and recording the path from the original node. The receiving nodes transmit the current built routing path, using the `RREP` messages, in the opposite direction towards the original node.

Similarly to AODV, when a link on the routing path is broken the `RERR` message is used to report the problem to the source node, which reinitiates the routing discovery.

2.4.3 Hybrid Routing Protocols

Hybrid routing protocols for MANETs appeared as a way of answering to the limitations of reactive and proactive protocols. Depending on the target scenarios, hybrid protocols can combine the benefits of both proactive and reactive protocols in a balanced way.

2.4.3.1 Zone Routing Protocol (ZRP)

The Zone Routing Protocol (ZRP) [Haas et al., 2002] is a hybrid routing protocol that combines several advantages from both proactive and reactive protocols. The ZRP protocol can be considered a framework as it lays on three internal protocols for route discovery and maintenance: the Intra-zone Routing Protocol (IARP), the Inter-zone Routing Protocol (IERP) and the Bordercast Resolution Protocol (BRP).

ZRP divides the entire network into variable size, overlapping zones, in order to avoid the typical overhead caused by hierarchical partitions. Thus, the network can be considered as flat, providing route optimisations as the nodes overlap their zones, fact that is very common on MANETs.

In ZRP, each zone contains at least one special node, namely the peripheral node, which is responsible to forward the traffic between its zone and others. The IARP protocol is responsible for acquiring the routes to the neighbours of each node as well as to their peripheral nodes, performing this discovery in a proactive manner. The communication between the several zones in the network is guaranteed by the IERP protocol, routing the messages only between peripheral nodes. Finally, the BRP protocol is used by the ZRP to transport the routing control messages exchanged by the IERP between peripheral nodes.

2.4.3.2 Zone-based Hierarchical Link State (ZHLS)

In the Zone-based Hierarchical Link State (ZHLS) [Joa-Ng and Lu, 1999] hybrid routing protocol, the network nodes are aware of their geographical location, by the assistance of locating systems, such as the GPS. Contrary to the ZRP protocol, the network is partitioned into several non-overlapping zones based on geographical information, relying on a hierarchical scheme.

Each zone of the network is assigned with a zone ID and its nodes assigned with a node ID. Every node is capable of retrieving its zone according to its physical location, as the global zone map is known by all the nodes inside the network. Considering the fact that ZHLS only has two levels of hierarchy, two types of routing updates can be performed: the zone level `Link State Packet (LSP)` and the node level `LSP`. A node level `LSP` contains all the node Identifications (IDs) of its zone and the zone IDs of all zones existent in the network. The node level `LSP` updates are periodically exchanged between nodes within the same zone. In case of connection breaks, a zone level `LSP` update is transmitted through the entire network, so that all nodes know the current global map of the network.

The routing information is maintained on two separate tables, one meant for intra-zone and the other for inter-zone routes. Before packet transmission, the source nodes first check their intra-zone routing table. If the destination node is located in the same zone the packet is directly delivered to its target, otherwise a location request is sent to all zone gateway nodes of the network. The correct gateway node (i.e. the gateway of the zone on which the destination is located) sends a location response to the source node, containing its zone ID. Finally, the source node, once it has retrieved the zone ID of the destination, transmits the necessary data.

2.5 Cluster-based Routing Protocols in Mobile Ad-hoc Networks

In contrast with regular routing protocols, the cluster-based routing protocols create or rely on an hierarchical network topology. Such protocols perform hierarchical routing between clusters, typically electing a cluster leader (clusterhead) and utilise gateway nodes for inter-cluster communication. The most relevant cluster-based routing protocols are described focusing on their clustering techniques.

2.5.1 Proactive Routing Protocols

Most existent cluster-based routing protocols are reactive, however there are a few solutions that provide proactive cluster-based routing.

2.5.1.1 Clusterhead Gateway Switching Protocol (CGSR)

The Clusterhead Gateway Switching Protocol (CGSR) protocol [Ching-Chuan et al., 1997; Cheng et al., 2013] is a combination of two existing schemes, the LCC clustering algorithm and the DSDV routing protocol. As previously studied in Section 2.3.1, in LCC clusterhead re-elections only occur if a change in the network causes two clusterheads to come into one cluster or if one node looses connectivity to all clusterheads. This provides resistance to cluster changes due to high dynamic networks and therefore increased stability. CGSR operates in the following manner. The source node transmits the packet to its clusterhead, which is forward to the gateway node that connects to the next clusterhead according to the destination route. The gateway node delivers the packet to the next clusterhead and the process is repeated until the destination cluster is reached. Finally, the destination clusterhead transmits the packet to the destination node. Each node broadcasts its clustermember table periodically using the DSDV protocol.

2.5.1.2 Cluster-based OLSR (C-OLSR)

The Cluster-based OLSR (C-OLSR) [Ros and Ruiz, 2007b] is an extension of the OLSR protocol providing routing to a clustered network topology. C-OLSR does not have clustering mechanisms, it only assumes that somehow the network is partitioned into clusters and restricts the propagation of `Topology Control (TC)` messages inside each cluster. The generation of inter-cluster routing information is based on Multipoint Relays (MPRs) at the cluster level, whereas each cluster is treated as a super node. the `Cluster HELLO (C-HELLO)` and `Cluster Topology Control (C-TC)` messages are responsible for the maintenance of routing information between clusters.

The C-OLSR protocol may be regarded as two instances of OLSR running simultaneously, the first being responsible for intra-cluster and the second for inter-cluster routing maintenance.

2.5.1.3 Deferred Routing (DefeR)

The Deferred Routing (DefeR) scheme [Palma and Curado, 2012] provides a novel approach to efficiently handle message forwarding under clustered networks. Driven to reduce the impact of node mobility, it defines virtual clusters organised in a binary tree of multiple levels. While the real clusters are managed by an external clustering algorithm, virtual clusters are defined by DefeR. A possible example is illustrated in Figure 2.2. The real clusters are always located at the leafs of the tree (in this case, the 7, 8, 9, 10, 13 and 14 are the actual clusters) and the virtual clusters are represented by the branches. The usage of this virtual hierarchy allows the routing protocol to be more resilient to node mobility, thus improving routing scalability. For instance, if a node moves from cluster 10 to cluster 9, the update of routing tables is limited to the nodes belonging to virtual cluster 4. This mechanism provides an abstract view of the network with different levels of granularity, in order to minimize the routing overhead.

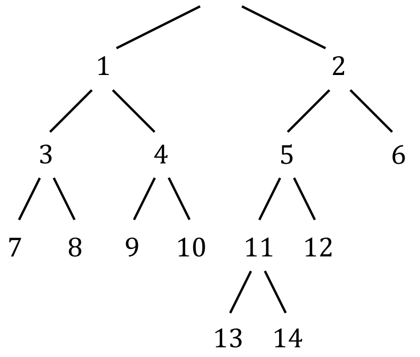


Figure 2.2: Example of a DefeR Hierarchy Tree

2.5.1.4 Summary of Cluster-based Proactive Routing Protocols

Table 2.8 shows the presented routing protocols. The C-OLSR protocol alone does not provide a clustered structure, however it provides an hierarchical routing topology compatible with any clustering scheme.

Table 2.8: Summary of Cluster-based Proactive Routing Protocols

Protocol	**C-Type**	**CH**	**Hops**	**OL**	**Extended Schemes**	
					Clustering	**Routing**
CGSR	ID-Neighbour	Yes	1-Hop	Yes	LCC	DSDV
C-OLSR	–	–	–	–	–	OLSR
DefeR	(vc) Location	(vc) No	(vc) n-hop	(vc) No	–	OLSR messages

C-Type: Clustering type
CH: Clusterhead-based clustering
Hops: Maximum number of hops inside clusters. If clusterhead-based, between any node and the clusterhead. Otherwise, between any two nodes
OL: Overlapping clusters
(vc): Annotation referring to virtual clusters

2.5.2 Reactive Routing Protocols

Cluster-based reactive routing protocols perform source route discovery on-demand, even tough the cluster structure is often created during the network start up. The studied cluster-based reactive routing protocols are presented as follows.

2.5.2.1 Core Location-Aided Cluster-based Routing Protocol (CLACR)

In Core Location-Aided Cluster-based Routing Protocol (CLACR) [Shih and Yen, 2006] the entire network is partitioned into square, non overlapping, clusters. Each cluster is assigned with a clusterhead responsible for routing. When the network is initiated and partitioned, each node assigns itself as a clusterhead and broadcasts a *HEAD_INIT* packet advertising its cluster ID and location, which is obtained by GPS. Nodes in the same cluster, upon receiving this packet, compare their location with the sender. If the sender location is nearer to the geometric centre of the cluster, then it becomes its current clusterhead.

CLACR is a reactive routing protocol. Instead of blindly flooding route requests throughout the network, it computes the desired route using Dijkstra algorithm in a cluster-by-cluster basis. Only clusterheads, source and destination nodes are needed to participate in route discovery, since clusterheads are directly linked (1-hop distance) and the maximum distance between the clusterhead and clustermembers is 1-hop. Despite the significant reduction of overhead, CLACR does not provide multi-hop routing at a hierarchical level. Instead, the entire network must be divided into a large amount of small clusters in order to operate.

2.5.2.2 Cluster Based Routing Protocol (CBRP)

The Cluster Based Routing Protocol (CBRP) divides the network into overlapping or disjoint clusters. Direct communication between clusterheads is not allowed. The communication between clusterheads is relayed through gateway nodes, which have in their range two or more clusterheads as neighbours (when clusters overlap) or have at least one clusterhead and another gateway node (when clusters are disjoint). As a result, CBRP forms clusters with a maximum distance of 2 hops. CBRP uses the Least Cluster Change (LCC) algorithm whereas the nodes with the lowest ID in a neighbourhood are elected as clusterheads.

The route discovery is performed as source routing by flooding the network with `RREQ` messages. These messages are passed between clusterheads and gateway nodes, however the latter do not broadcast them, they only serve as relay nodes to the next clusterhead. This process efficiently minimises the amount of flooding traffic during route discovery while also reducing the delay of route discovery.

More recently, several improvements have been proposed for CBRP. The Cross Layer Cluster Based Routing Protocol (Cross-CBRP) [Jahanbakhsh and Hajhosseini, 2008] exploits the Physical and Media Access Control (MAC) layers to better adapt the clustering algorithm to link variation due to mobility. Instead of electing clusterheads using the lowest ID, the Cross-CBRP implements a new relative mobility metric which measures the mobility levels, energy levels and signal strength between neighbours in order to elect better suited clusterheads.

The Efficient Cluster Based Routing Protocol (ECBRP) [Yu et al., 2008] also proposes enhancements to CBRP. In CBRP cluster changes occur frequently and the clusters tend to be highly overlapped [Yu and Chong, 2006]. In order to solve this problem, the ECBRP uses a different clustering scheme, namely Efficient Clustering Scheme (ECS) [Yu and Chong, 2006], instead of LCC. ECS is also based in the Lowest ID, however with significant modifications. A node will become a clusterguest when it looses connectivity with its clusterhead but still is connected to at least one clustermember belonging to any cluster. This mechanism avoids the formation of new clusters. Furthermore, when a clusterhead perceives that all its clustermembers are covered by other clusterheads it invokes a cluster deletion process, becoming the clusterhead itself a clusterguest. This way, the cluster overlapping is highly reduced. The routing process of ECBRP is identical to CBRP.

The Cluster Based Trust-aware Routing Protocol (CBTRP) [Safa et al., 2009] is an extension of CBRP providing security mechanisms. CBTRP protects the forwarding of packets to reach intermediary malicious nodes. Clusterheads are replaced as soon as they become malicious and automatically update routes to avoid malicious nodes. The *trust* of a node is determined by the analysis of received, forwarded and overheard packets. A positive feedback can be attained if a node forwards packets within a certain time window or/and generates successful replies. A negative feedback can be generated if a node refuses to forward packets (to save energy or due to simple malicious behaviour), if it tampers packet data or it forwards route requests abnormally. In CBTRP each node monitors the behaviours of its neighbours, assigning a *trust* value to each. If *trust* becomes inferior to a certain threshold, the node is immediately reported and removed from calculated routes.

2.5.2.3 On-Demand Clustering Routing Protocol (OCRP)

To build and maintain an hierarchical topology, existing clustering mechanisms consume a significant amount of communication overhead. The On-Demand Clustering Routing Protocol (OCRP) [Huang et al., 2006] introduces a novel paradigm in the way clusters are created and maintained. The phase of clustering is eliminated and directly merged in data transmission, i.e. clusters are created impromptu when nodes wish to send data. OCRP introduces a full reactive paradigm, clustering topology and route discovery is only performed once nodes transmit data.

When a source node starts a flood search from a cold network start, every receiving node is ready to be a clusterhead. The first neighbour which relays the flooding packet becomes the clusterhead, and all the other nodes which receive the first relay become ordinary nodes. Once another node, from the same neighbourhood, becomes a clusterhead it turns into a gateway, this way defining the cluster boundary. Upon data arriving at the destination, all the nodes in the path belong to the cluster structure.

2.5.2.4 On-Demand Routing-based Clustering (ORC)

Similarly to OCRP, the On-Demand Routing-based Clustering (ORC) scheme [Hsu and Feng, 2007] also creates the cluster structure and discovers routing paths on-demand, however with additional clustering maintenance. While in OCRP clusters are created only by exploiting the `RREQ` messages, ORC relies on additional `Cluster-RREQ (C-RREQ)` packets to create the cluster topology. A `C-RREQ` packet is a variation of the `RREQ` packet and it is created when a `RREQ` packet is received by a node. It contains additional information, such as the Cluster-ID (CID) and the ID of neighbour nodes. This information allows ORC to support multi-hop clustering with overlapping clusters.

2.5.2.5 Summary of Cluster-based Reactive Routing Protocols

Table 2.9: Summary of Cluster-based Reactive Routing Protocols

Protocol	**C-Type**	**CH**	**Hops**	**OL**	**Extended Schemes**	
					Clustering	**Routing**
CLACR	Location	Yes	1-Hop	No	–	–
CBRP	ID-Neighbour	Yes	1-Hop	Yes	LCC	–
Cross-CBRP	Combined-weight	Yes	1-Hop	No	–	CBRP
ECBRP	ID-Neighbour	Yes	2-Hops	Yes	ECS	CBRP
CBTRP	ID-Neighbour	Yes	1-Hop	Yes	CBRP	CBRP
OCRP	On-Demand	Yes	1-Hops	No	–	–
ORC	On-Demand	Yes	n-hops	Yes	–	–

C-Type: Clustering type
CH: Clusterhead-based clustering
Hops: Maximum number of hops inside clusters. If clusterhead-based, between any node and the clusterhead. Otherwise, between any two nodes
OL: Overlapping clusters

Table 2.9 shows the presented routing protocols. The cluster-based reactive routing protocols only perform route discovery on-demand. However, most of these protocols pre-establish a clustered structure before any route request takes place. Amongst the presented contributions, only OCRP and ORC schemes are entirely reactive, also able to create the cluster structure on-demand, simultaneously with the route discovery.

2.6 Summary and Open Issues

As studied in this chapter, there is a large amount of solutions aiming to deliver scalability to MANETs. An hierarchical topology is the most feasible paradigm to allow these networks to scale, either provided by a clustering scheme alone or with the usage of a cluster-based routing protocol. Both approaches have advantages and drawbacks, depending on the application.

A clustering scheme can be regarded as an independent network management mechanism, providing two layers of topology in which distinct routing protocols can operate. For example, given a scenario where short distance communications are realised more often than long distance, a proactive routing protocol can be assigned to handle intra-cluster communications and a reactive routing protocol to handle inter-cluster communications. Thus, a clustering scheme independent from routing is more flexible than a cluster-based routing solution, allowing a multiple combination of solutions and paradigms. On the other hand, the combination of different routing protocols, operating at different topology layers, cannot be deployed spontaneously, as it requires a pre-configuration of the protocols to bridge the intra-cluster and inter-cluster communication. This is why, several routing protocols, such as C-OLSR and DefeR, are programmed to be aware of an hierarchical topology, even tough not being responsible to handle the clustering.

The great majority of clustering schemes available in the literature rely on central nodes, i.e. clusterheads, to manage the cluster structure. The main reason of this fact is to reduce the complexity of the clustering algorithm. By entrusting cluster management decisions to a unique node, remaining nodes are not required to reason between decisions, thus reducing communication overhead. However, this gain is debatable, as clusterheads also bear many disadvantages. The election of clusterheads is a complex and costly process. Thus, as discussed in this chapter, the main concern of clustering schemes is to provide a proper clusterhead election algorithm capable of selecting the most suitable nodes. Moreover, when clusterhead nodes fail, due to energy depletion, lost connection or simple device shutdown, a clusterhead re-election process must also be performed. This process often triggers a complete re-formation of the cluster at hand, leading to a high resource cost and possible packet losses. In fact, clusterheads are more prone to fail due to low energy than remaining nodes, due to the extra data processing and communication. Finally, clusterheads typically represent bandwidth congestion points, since all traffic must be forwarded through these nodes.

Chapter 3

Clustering for Multi-purpose Scenarios

One of the most critical limitations of Mobile Ad-hoc Networks is the scalability of routing protocols. In order to establish an impromptu network infrastructure capable of managing hundreds of mobile nodes, routing protocols struggle to deliver updated routes at any moment. Clustering schemes support routing protocols by providing an hierarchical network structure, dividing the network into small groups to empower routing scalability.

In this chapter, a new clustering scheme is presented in order to deliver scalability in MANETs. The Smart and Balanced Clustering (SALSA) scheme is introduced, describing its design, algorithm specification and a formal analysis of its complexity. An evaluation is also presented, especially focusing SALSA performance in generic, multi-purpose scenarios.

3.1 Design and Specification

Most of previous clustering schemes have relied on the utilisation of central nodes, or clusterheads, to create and maintain a cluster structured network. The employment of these nodes brings several drawbacks when considering network performance. First, the clusterhead election algorithm is typically a complex and time-consuming process, being executed once a clusterhead is no longer suitable or available. As a result of this process, multiple re-clustering operations are performed in order to update the information of the new cluster leader, therefore causing instability and control overhead. The second drawback of clusterheads is related with their role. Clusterheads have the responsibility of routing all intra-cluster traffic. Additionally, they are also required to forward inter-clustering traffic in case of its cluster being in the path. Thus, in most cases, clusterheads represent bandwidth bottlenecks in the network. Another drawback of clusterheads concerns a fast energy depletion of such nodes, often forcing clusterhead re-elections which introduce more additional overhead.

The SALSA approach attempts to provide a spacial usage of bandwidth where clusters are more robust to topological changes. SALSA is a distributed and self-organising clustering scheme designed to operate in generic, multi-purpose MANETs. Its main purpose is to build stable clusters aiming to significantly reduce the control overhead, thus providing a light hierarchical structure for routing. This proposal is designed to build a cluster topology in a distributed fashion, meaning that each node in the network will have the same role, not relying on centralised nodes, like clusterheads.

SALSA introduces a new load-balancing algorithm, which acts progressively along time. During execution, the size of clusters is monitored allowing nodes to be distributed across them, in order to maintain well balanced clusters. Before the maximum capacity of a cluster is reached, it starts to assign nodes to neighbour clusters or, in cases where this operation is not possible, builds a new cluster to receive excess nodes.

It was also designed to reduce the clustering control overhead. This objective was mainly accomplished by utilising small and purpose-driven specific messages. As a result, five different types of messages are utilised, aiming to decrease the amount of transmitted traffic.

3.1.1 Node States

In SALSA, nodes can be in one of three distinct states, namely *Unclustered*, *Clustered* and *Clustered-GW*, as shown in Figure 3.1.

The *Unclustered* state typically represents a temporary role, as the node is waiting to be assigned to a cluster. In this state, when the node discovers neighbours, it waits a predefined period of time in order to calculate the best candidate cluster to join.

Nodes in the *Clustered* state usually represent the majority of nodes in the network, whereas all in-range nodes must belong to their cluster. Thus, the communication with foreign nodes (i.e. nodes assigned to a different cluster) is performed through gateway nodes.

Finally, the *Clustered-GW* state is assigned to nodes that have in-range foreign nodes, i.e. they must have direct connectivity with at least one different cluster. Thus, they are responsible of forwarding inter-cluster maintenance messages and typically are located on the edge of clusters.

3.1.1.1 State Transitions.

The *Unclustered* state occurs on two different situations:

1. Node isolation - in this case the node does not have any in-range neighbour nodes, therefore cannot create or be assigned to a cluster.

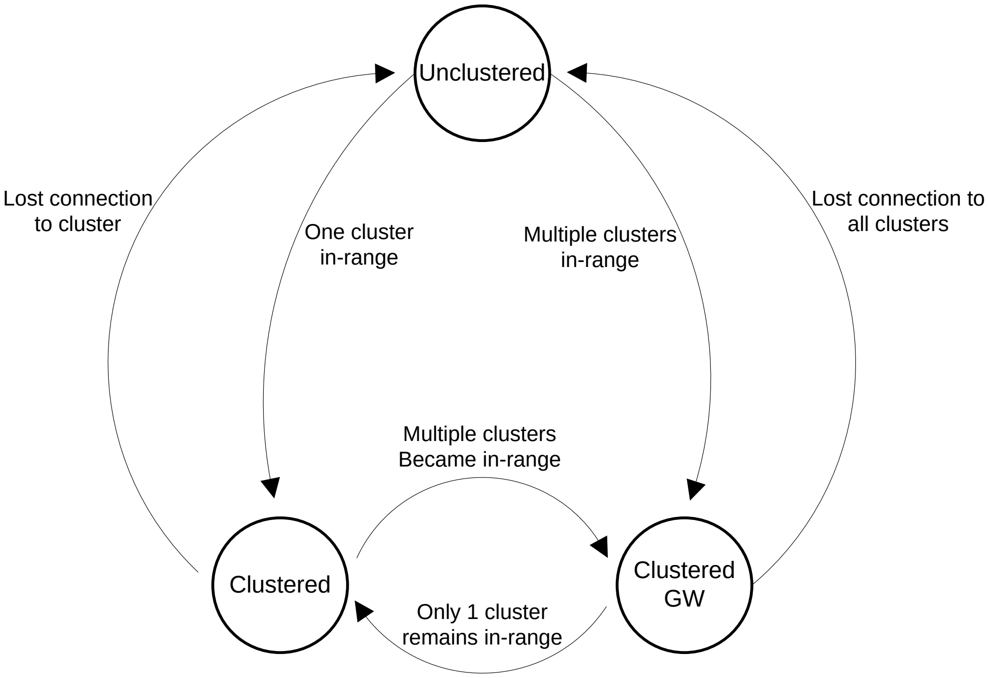


Figure 3.1: Node states

2. Cluster transition - the management of clusters occasionally requires nodes to change clusters, due to cluster balancing. In this phase, nodes can be unassigned from a cluster.

Unclustered **to** ***Clustered*** This state transition occurs when a node becomes aware of an in-range cluster or an unclustered node. In the first situation, the node joins the cluster automatically. However, if the node only detects unclustered nodes, a new cluster is created to adopt the unclustered nodes.

Unclustered **to** ***Clustered-GW*** This transition is similar to the previous, but more than one cluster is discovered. Firstly, the node calculates which is the best, taking several parameters into account: number of in-range nodes for each cluster and the size of clusters. The greater the number of in-range nodes, the stronger connection to the cluster. However, if the size of the cluster is high, possibly close to the maximum allowed, this cluster would be a bad choice. To measure this trade-off, a new metric is utilised (3.1), namely the best clustering metric (BC), where BC_i is the metric value for cluster i.

$$BC_i = AP_i + \frac{IRN_i}{2} \tag{3.1}$$

AP_i is defined as the number of the available positions in cluster i until it reaches the maximum allowed, i.e. the difference between the maximum allowed number of nodes per cluster and the current number of assigned nodes. IRN_i is the number of in-range nodes belonging to the cluster. As a result, the cluster with the highest BC value is chosen by the node.

Clustered* to *Clustered-GW This transition occurs when a node becomes aware of clusters, excluding its own.

Clustered-GW* to *Clustered Whenever a clustered gateway node loses connection with all its foreign clusters, it automatically transits to a normal clustered state.

Clustered/Clustered-GW* to *Unclustered A node becomes unclustered when willingly disconnects from the network or loses connection with all its neighbour nodes. When this situation occurs, it is necessary to verify the consistency of the cluster to guarantee that all home nodes (nodes assigned to the cluster) can communicate with each other.

3.1.2 Maintenance Information and Messages

This subsection describes the information that each node maintains and the messages utilised. There are two tables providing insight of the network topology, namely the *NODE_TABLE* and the *CLUSTER_TABLE*. The *NODE_TABLE* keeps all the information about neighbour and home nodes, as described in Table 3.1.

Table 3.1: Node maintenance information

Information	**Description**
Node-ID	Unique identifier of the node
State	Current state of the node (*Unclustered*, *Clustered* or *Clustered-GW*)
C-Degree	Value to determine the connection type towards this node. Value ranges from 0 to 5, whereas 0 represents a non-neighbour (therefore merely a home node), 1 denotes a lost connection towards this node and finally, 2-5 values represent the quality of the connection, being 5 the best possible connection.
Alive	Boolean value, determining whether the node is responding or not

SALSA relies on multiple, small purpose-driven, messages to manage the cluster structure. All messages contain one common field, *Type_ID*, which uniquely identifies the message type that is being transmitted. Apart from this field, all the messages contain different sets of fields, suitable to their purpose, as follows:

- *Ping* - periodic broadcast message, allowing nodes to discover their neighbourhood.
- *Hello* - provide the structure of the cluster to member nodes.
- *Lost Hello* - broadcasted when a node loses connection with a neighbour home node, informing member nodes, that do not have direct connection, about a possible disconnected node. This event triggers a process in order to verify if the node is still connected via other nodes, namely the alive check process. At the end of this process, if it is verified that the node is in fact disconnected, it is necessary to verify if the cluster is still consistent, which implies the utilization of the following described message (*Alive Hello*).
- *Alive Hello* - upon the trigger of an alive check process, to verify the consistency of the cluster, i.e. guarantee that all nodes inside the cluster are capable of communicating with each other. In most situations the cluster remains consistent; however there are cases in which the cluster becomes partitioned in two clusters. In this particular situation, both clusters have the same identifier, thus it becomes imperative to change it.
- *Switch Hello* - used when a cluster identifier becomes inconsistent and it is necessary to change the *Cluster_ID* for their nodes. This message is triggered at the end of the alive check process when nodes detect that the cluster is partitioned, thus being necessary to change the cluster identifiers.

3.1.3 Algorithm Specification

The operation of the clustering algorithm differs according to the node state. The main operation logic for the unclustered and clustered states are defined as follows.

3.1.3.1 Unclustered state

In the unclustered state only two types of messages are accepted, namely *Ping* and *Hello*. Upon receiving a *Ping* message, the node must check if the sender node is clustered. If so, it starts a specific timer to join the cluster structure. Otherwise, it starts a different timer, in order to create a new cluster. Before these timers expire, the node will potentially receive other *Ping* messages from distinct nodes, therefore the timer for cluster creation may be canceled before it expires. In fact, upon the initialisation of the cluster join timer, the cluster creation timer is automatically aborted, since there is no need to create a new cluster.

If an *Hello* message is received in this state, the node simply updates its *NODE_TABLE* structure, according to the information carried by the message. Naturally, only *Hello* messages received from clustered nodes are accepted.

Algorithm 1 Receive Message - *Unclustered*

```
if received PING then
    update NODE_TABLE
    update CLUSTER_TABLE
    if sender node is clustered then
        init cluster_join_timer
    else
        init cluster_create_timer
    end if

else if received HELLO and sender node is clustered then
    update NODE_TABLE
    update CLUSTER_TABLE
end if
```

3.1.3.2 Clustered state

In the clustered state, all types of messages are accepted. Once a clustered node receives a *Ping* message from a clustered node, it must check if its *NODE_TABLE* contains the sender node. If it does not, it means that the sender node is a new member of the cluster structure, and an *Hello* message must be sent, in order to provide information about the home nodes of the cluster.

The remaining types of messages are only accepted if they are sent by a home node, i.e. a node belonging to the same cluster. If and *Hello* message is received, the maintenance tables are updated and the message is forwarded. If a *Lost* message is received, it means that some node in the cluster lost connection with another. In this case, each receiving node considers all nodes in the cluster as dead, and only otherwise upon the reception of an *Alive* message from each one. Thus, when a *Lost* message is received, all nodes in *NODE_TABLE* are set to dead and a request to check their status is sent, by the broadcast of an *Alive* message. Likewise, upon the reception of an *Alive* message, the sender node is set to alive. This whole process is required for failsafe purposes, since in some cases a single home node is vital for the cluster stability. For instance, if a cluster has the geometric shape of an hourglass, in which only one node is located in the middle, connecting the cluster, the cluster becomes incoherent if it looses connection with that particular node.

When a node receives an *Switch* message, which may be triggered by any node at the end of the alive check process (Section 3.1.2), the node rapidly changes its cluster to the one

Algorithm 2 Receive Message - *Clustered*

```
if received PING then
    update NODE_TABLE
    update CLUSTER_TABLE
    if sender node is new clustered then
        broadcast HELLO
    end if

else if received HELLO and own_cluster_ID equals sender_cluster_ID then
    update NODE_TABLE
    update CLUSTER_TABLE
    forward HELLO

else if received LOST HELLO and own_cluster_ID equals sender_cluster_ID then
    forward LOST HELLO
    set all dead in NODE_TABLE
    broadcast ALIVE HELLO
    init alive_timer

else if received ALIVE HELLO and own_cluster_ID equals sender_cluster_ID then
    forward ALIVE HELLO
    if alive_timer is inactive then
        set all dead in NODE_TABLE
        init alive_timer
    end if
    set sender alive in NODE_TABLE

else if received SWITCH HELLO and own_cluster_ID equals sender_cluster_ID then
    update NODE_TABLE
    update CLUSTER_TABLE
    $own\_cluster\_ID \leftarrow switch\_cluster\_ID$
    forward SWITCH HELLO
end if
```

specified in the message and the maintenance tables are updated according to the information received. Also, in this case, the message is forwarded to remaining nodes.

3.1.4 Complexity Analysis

In this section, an analysis of the overhead introduced by SALSA is performed. The scheme operations can be classified as follows:

- Overhead due to *Ping* messages (OH_{Pg})
- Overhead due to Cluster Formation (OH_{CF})
- Overhead due to Cluster Maintenance (OH_{CM})

As previously described, SALSA utilises a *Ping* message mechanism so nodes are able to discover their neighbourhood. Thus, since the broadcast of these messages is constant during the execution of the algorithm, it must be analysed aside from the remaining operations. The network model of SALSA for the analysis of the clustering overhead relies on the following parameters:

- N = the number of nodes in the entire network
- M = predefined constant, defining the maximum allowed number of nodes per cluster
- t_{ping} = predefined period of time for *Ping* message broadcast
- $t_{formation}$ = predefined period of time for initial cluster formation (join or cluster creation)
- t_{join} = predefined period of time for node join operation
- t_{change} = predefined period of time for node cluster change
- t_{alive} = predefined period of time to determine status of nodes

3.1.4.1 Ping Overhead

In SALSA, *Ping* messages are broadcasted periodically. This process implies an overhead of $t_{ping}N$ messages per time step. Since t_{ping} is a predefined constant by the algorithm, the overhead of the *Ping* message is $OH_{Pg} = O(N)$ per time step.

3.1.4.2 Cluster Formation Overhead

In the cold start of SALSA, where all nodes in the network are unclustered, each node waits a predefined period of time, whether to create a new cluster or to join a recently created one. Thus, before a node being assigned to the cluster structure, it must wait at least a $t_{formation}$ period of time. Following this procedure, several *Hello* messages will be broadcasted to it, providing the necessary information about its cluster. The number of *Hello* messages broadcasted is equal to the number of 1-hop neighbours of the node, inside its cluster. Thus, in a worst case scenario, a recently clustered node will receive M *Hello* messages. The *Hello* messages are broadcasted within a constant time period, and therefore it takes only 1 time step for this process, which adds up to $t_{formation} + 1$. Analysing the complexity for the entire network, the overhead is $(t_{formation} + 1M)N$, resulting in $O(MN)$. However, since the formation process only occurs once during the entire execution, and not constantly as for *Ping* messages, the total overhead is $OH_{CF} = O(1)$.

3.1.4.3 Cluster Maintenance Overhead

The maintenance of clusters is divided in two main routines, namely the joining of a new node and the leaving of a node. These two events are responsible for triggering all the operations to manage the cluster structure.

Joining of New Node When a node joins a cluster two operations may be triggered, namely the auto-balancing of clusters or the creating of a new cluster, due to the imposed maximum nodes per cluster. In most cases, the node simply joins a cluster without requiring these operations, however for the complexity analysis, the worst case scenario must be considered. When a node wishes to join the cluster structure, it waits a predefined period of time t_{join} in order to discover the neighbourhood, and to choose the most suitable cluster. Upon choosing its cluster, the node assigns itself to it and receives an *Hello* message from a member node, similarly to the initial phase of cluster formation. Thus, the join operation alone has a complexity of $t_{join}N$ for the entire network, and an overhead of $O(N)$ per time step.

The auto-balancing mechanism may be triggered once a node joins a cluster, which requires a node to be assigned to a different cluster. In this process, the node waits a random amount of time, no longer than a predefined period t_{change}. When this time expires, the node emits an *Hello* message, informing its former members that it is no longer assigned to that cluster. This process implies a time complexity of $(t_{change}+1)N$ which results in an overhead of $O(N)$ per time step.

The creation of a new cluster is also an operation that can be triggered by the join cluster operation, when auto-balancing is not possible. This operation, is executed before the new node joins the cluster. Since the operation does not affect the topology of existing clusters, the message complexity does not exist, since the existence of the new cluster is broadcasted using *Ping* messages. In short, it will only cost the period of time t_{join}, resulting in an overhead of $O(1)$ per time step. Summarising, the overhead of joining of new node is $O(N)+O(N)+O(1)$ which results in $O(N)$.

Leaving of a Node When a clustered node detects that it has no longer connection to one of its member neighbours, it broadcasts a *Lost_Hello* message. Upon the reception of this message, each node waits a predefined period of time t_{alive} and broadcasts an alive message. This process results in a message complexity of $t_{alive}MN$, which implies an overhead of $O(N)$, since M is a constant predefined by the algorithm. After this process, as the cluster may lose its consistency, a *Switch_Hello* message is broadcasted to build two new clusters. In the worst case scenario, M messages are broadcasted, resulting in complexity of $(t_{alive}M+M)N$, with an overhead of $O(N)$.

Total Maintenance Overhead As analysed above, the overhead of joining of a new node is $O(N)$ and the leaving of a node is also $O(N)$, which results in a maintenance overhead of $OH_{CM} = O(N)$.

3.1.4.4 Total Clustering Overhead.

Summarizing this analysis, the total overhead is denoted by $OH_C = OH_{Pg} + OH_{CF} + OH_{CM}$, which results in $OH_C = O(N) + O(1) + O(N)$. Consequentially, SALSA has a total clustering overhead of $O(N)$ per time step.

3.2 Performance Evaluation

To properly examine the effectiveness of SALSA, a simulation evaluation, driven by the main objectives of the scheme, was performed using the OPNET Modeler [OPNET Technologies Inc. (Bethesda USA), 1986]. Therefore, the main purpose of this simulation evaluation is to assess the stability and low overhead capabilities of SALSA. To accomplish this objective, a set of different simulation environments, featuring the network size and speed of nodes, were defined.

3.2.1 Environment and Parameters

The performance of clustering schemes is strongly influenced by the scenarios under which they are evaluated. For instance, a better performance is expected for low-density networks (i.e. low quantity of nodes per Km^2) or with nodes moving at low speeds.

The scenarios used for SALSA evaluation were selected in such a way that they represent, as much as possible, realistic scenarios. For this specification, the evaluation parameters can be divided in two groups, the fixed-value and the variable-value parameters, according to whether their value changes for different simulation scenarios (Table 5.1).

Given the enormous quantity of different possible scenarios that the combination of parameters provides, only the most significant were chosen. In particular, the parameters that most influence the scalability of the network are the network size (number of nodes) and the maximum speed that nodes can achieve.

Considering the vast application that clustering can have and that this simulation study aims to evaluate a generic, multi-purpose scenario, a specific node mobility pattern, like Group Mobility, Freeway or Manhattan models would not be suitable [Divecha et al., 2007]. Thus, a random model, the Random Waypoint, was preferred. Also, for a simulation of 900 seconds, a 50 second pause time was chosen. Each simulation execution was repeated 30 times, assigning to each a distinct seed value.

Table 3.2: Simulation parameters

Fixed-value parameters	
Simulator	OPNET Modeler 16.0
Field Size (m^2)	5000×5000
Node mobility algorithm	Random Waypoint Model
Pause time (s)	50
Transmission range (m)	150
Bandwidth (Mbps)	11
Simulation time (s)	900
Maximum cluster size	50
Variable-value parameters	
Network size (number of nodes)	200; 400; 600; 800; 1000
Node maximum speed (m/s)	0; 5; 10; 15; 20

3.2.2 Results

This section presents the obtained results in the simulation. Several metrics were used to evaluate the effectiveness and performance of SALSA, individually described in the following subsections. The discussion of the obtained results briefly compares SALSA to A Novel Stable and Low-maintenance Clustering Scheme (NSLOC) [Conceição et al., 2010]. NSLOC is a previous proposal, also a distributed (clusterhead-free) scheme, which makes the comparison with SALSA appropriate.

3.2.2.1 Number of Clustered Nodes

This metric provides the average amount of nodes that are associated with the cluster structure, during the simulation execution and retrieved at one second interval. Nodes that are isolated, which are not in communication range with any other node, cannot be assigned to a cluster. Therefore, since the area of the scenario is constant for all network sizes, there is a bigger percentage or nodes that are likely to be unclustered in small networks.

Figure 3.2 and Table 3.3 show the percentage of clustered nodes for the different network sizes and node speeds, in SALSA and NSLOC respectively. The percentage of clustered nodes for large networks is bigger than for smaller networks. Naturally, this occurrence is strongly tied with the density of the network, since the probability of a node being in communication range with another is greater for networks with more nodes. For the static scenario, with a network size of 1000 nodes, around 94.4% of nodes are assigned to the cluster structure.

With the increase of node mobility this percentage is reduced to 85.9% with a node maximum speed of 20 m/s for the same network size. In comparison with NSLOC results, SALSA is more effective, capable of assigning a larger percentage of nodes to the cluster structure, particularly in larger networks. This slight difference is due to the node state transition specification, as it analyses the most suitable clusters based on the best clustering metric.

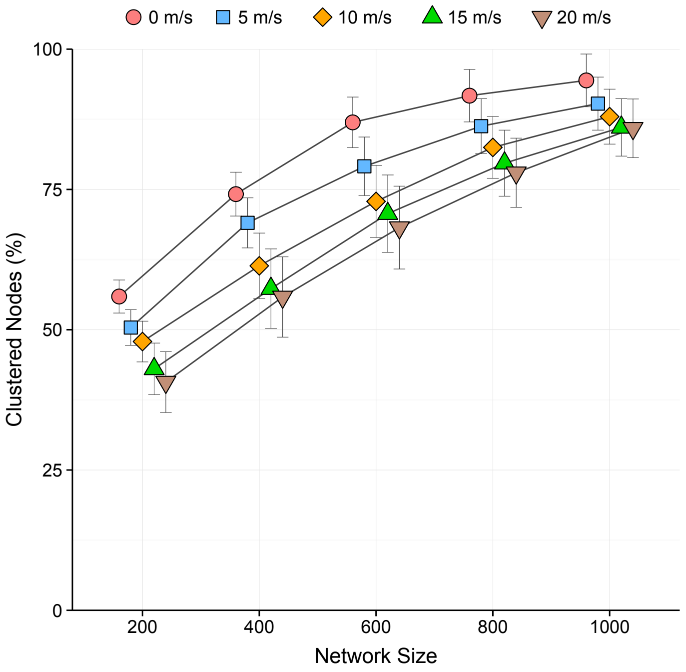


Figure 3.2: Amount of clustered nodes in SALSA (percentage)

Table 3.3: Amount of clustered nodes in NSLOC (percentage)

Size \ Speed	0	5	10	15	20
200	44.72	51.17	48.56	48.18	48.99
400	62.28	67.16	65.02	64.50	61.34
600	78.30	72.68	72.18	70.59	68.17
800	83.99	77.78	75.54	72.78	68.90
1000	87.41	80.79	79.50	74.96	72.91

3.2.2.2 Network Load

The network load represents the amount of transmitted traffic in the network. This metric translates the overall bandwidth usage, including the clustering control overhead.

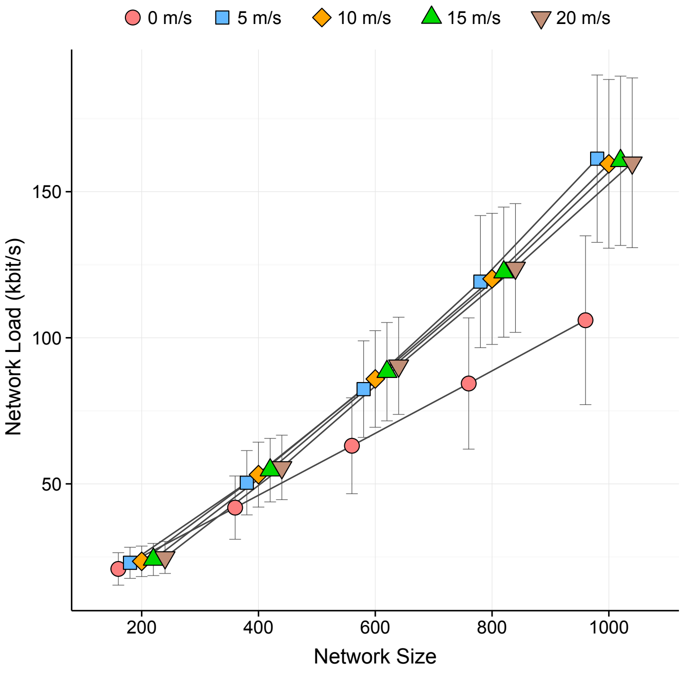


Figure 3.3: Total Network Load in SALSA

Figure 3.3 and Table 3.4 show the average of the total network load, for different velocities and network sizes, for SALSA and NSLOC respectively. As shown, the mobility of nodes has a significant impact in the clustering control overhead. Static scenarios present a lower network load, particularly in larger network sizes. The dynamic scenarios do not have a significant load difference. In fact, for the 1000 nodes network size, the scenario with a speed of 20 m/s

Table 3.4: Network Load in NSLOC (kbit/s)

Speed Size	0	5	10	15	20
200	198.798	215.938	228.984	235.012	250.668
400	216.073	242.593	262.297	261.015	276.367
600	214.386	260.947	286.995	300.862	312.435
800	237.937	287.703	304.135	332.680	364.699
1000	287.152	296.465	327.236	356.927	385.944

presents a slight lower load than remaining speed scenarios. This is explained by the lower number of connections (and hence, the lower amount of clustered nodes) inherent to higher node speeds.

NSLOC manages the cluster structure with a significantly higher network load. Contrary to SALSA, this scheme relies in only two types of messages, `PING` and `HELLO`, being the last used for all maintenance operations. This translates a bigger payload in each `HELLO` message, resulting in a higher network load. Generally, SALSA is capable of maintaining a clustered structure with low network resources, providing a feasible topology for routing protocols. In a 1000 nodes network, it uses an overall average of 160 $kbit/s$ per second, which represents an average load of 160 bit/s in each node.

3.2.2.3 Number of Messages

The number of messages required by the clustering scheme to operate increases with the size of the network.

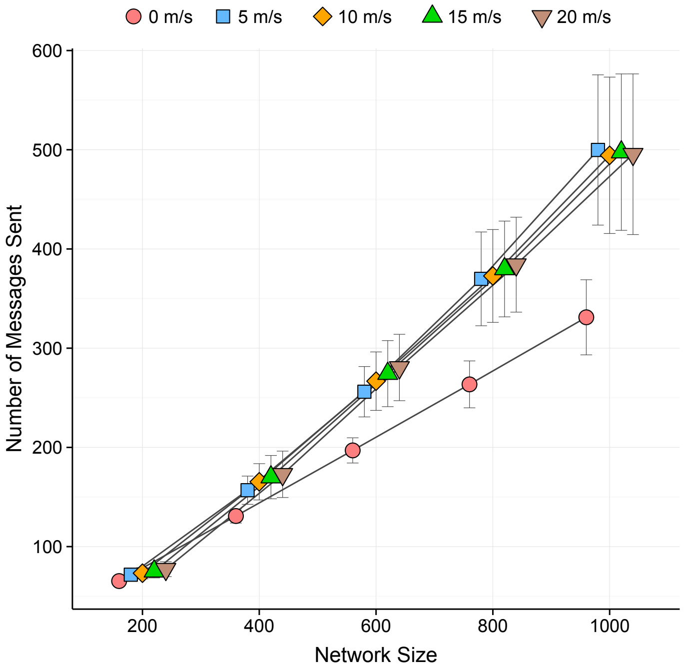


Figure 3.4: Average number of messages sent in SALSA

Figure 3.4 and Table 3.5 show the average number of control messages sent by each node, for the different network sizes and node speeds, in SALSA and NSLOC respectively. As expected, the shape of this chart is quite similar with the network load metric, since the number of messages sent is related with the overall network load.

The average number of messages sent by SALSA is not much lower than the ones sent by NSLOC, in spite of the significant difference in the average load results. This fact is due to the smaller and purpose-driven messages used in SALSA.

Table 3.5: Average number of messages sent in NSLOC

Speed **Size**	**0**	**5**	**10**	**15**	**20**
200	76.3	73.2	75.1	77.8	79.8
400	157.2	156.53	164.5	172.1	179.0
600	229.4	248.4	267.0	283.0	296.7
800	311.4	346.7	382.9	417.4	443.0
1000	453.2	451.3	507.1	570.5	614.4

3.2.2.4 Number of Stable Nodes

The stability of clusters can be measured according to the amount of time that nodes belong to a cluster, without suffering re-clustering operations.

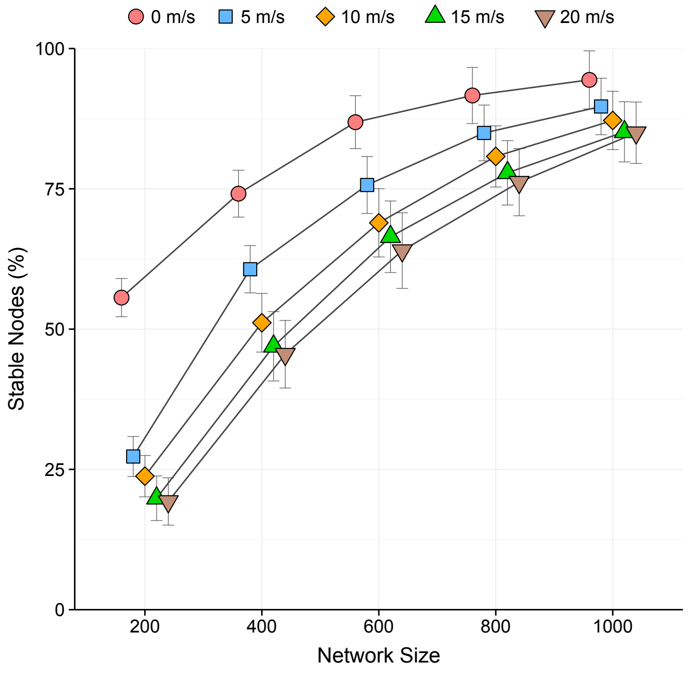


Figure 3.5: Amount of stable nodes in SALSA (percentage)

In other words, a node is considered to be stable if it remains active and assigned to the same cluster during a period of time. A stability metric (ST) is proposed to provide a mechanism of determining the amount of nodes that were stable during simulation for a period greater that the ST value. The stability time ST is defined in Equation 5.2.

$$ST = k \times \frac{r \times p}{v \times d} \tag{3.2}$$

r is the transmission range of nodes, p is the pause time, v the average of node speed (mean value of minimum and maximum speed), d the density of nodes (number of nodes per Km^2) and finally, k represents an arbitrary constant, equal in all simulation executions.

Figure 3.5 and Table 3.6 show the percentage of stable nodes per network size, at different node speeds, in SALSA and NSLOC respectively. The percentage of nodes represented in the chart were stable at least for a period of time equal to ST. As soon as a node suffers a re-clustering operation, the counter of stability time is reset and it is not considered stable until the ST time period passes.

Table 3.6: Amount of stable nodes in NSLOC (percentage)

Size \ **Speed**	**0**	**5**	**10**	**15**	**20**
200	38.47	21.51	17.87	16.86	15.76
400	55.20	35.37	30.83	31.19	27.45
600	71.85	42.80	42.79	40.08	35.13
800	78.49	51.94	46.63	39.46	36.72
1000	83.16	59.05	52.78	40.13	37.81

In the static scenarios, the percentage of stable nodes is identical to the percentage of clustered nodes (Figure 3.2). Since there is no mobility, nodes always remain in the same cluster throughout the simulation time. Dynamic scenarios show less stable nodes, particularly in low density networks. On the other hand, larger networks present a large amount of stable nodes, very close to the amount of clustered nodes.

3.3 Summary

In this chapter, the Smart and Balanced Clustering (SALSA) clustering scheme was presented and evaluated. By subdividing the control messages into smaller, specific purpose driven, messages the network load is significantly reduced in order to guarantee better scalability. Moreover, the cluster balancing mechanism and the best clustering metric allows to improve the stability of the cluster structure. The obtained results in SALSA evaluation confirm a large percentage of clustered nodes while maintaining the structure with a low overhead. SALSA is thus a suitable clustering scheme capable to provide a stable hierarchical structure to any type of routing protocol designed for MANETs.

The specification of SALSA resulted in an article entitled “Smart and Balanced Clustering for MANETs” published in the proceedings of “The 10th International Conference on Ad Hoc Networks and Wireless 2011”.

Chapter 4

Location-aware Clustering

One of the most important problems in the scalability of wireless ad hoc networks is related with the mobility of devices. The frequent connectivity failures in a cluster topology cause a high amount of maintenance tasks, such as cluster integrity verification and node re-assignments to the cluster structure. Location information can decrease the number of such tasks in order to improve the stability of these networks. Having the location information of each node in the network, clusters can be uniformly created with a balanced amount of members, promoting an even topology with potentially more stability.

In this chapter, a brief study of the most relevant location sensing solutions, focusing wireless ad hoc networks, is presented. Two new location-aware clustering schemes are introduced aiming to improve the stability of the network, namely the Clustering for Indoor and Dense Networks (CIDNET) and Adaptive and Location-aware Clustering (DiLoC). Finally, a performance evaluation is conducted, assessing the obtained advantages when using location information.

4.1 Location Sensing

In recent years, a wide growth of wireless systems has been noticed. Wireless technologies are present in consumer applications, medical, industrial, public services, transports and much more. Therefore, there is a high demand for accurate positioning in wireless networks, either for indoor or outdoor environments. Concerning the nature of the application, different types of location are needed, which can be characterised as physical location, symbolic location, absolute location and relative location. Physical location is expressed in coordinates, identifying a point on a map. Symbolic location refers to a location in natural language, such as a coffee shop, office, etc. Absolute location uses a global shared database system, which references all located objects. Finally, relative location is usually based on the proximity of devices, e.g. known reference points, providing an environment-dependent location. The latest is the most common used paradigm.

4.1.1 Measurement Techniques and Concepts

The main challenge of location estimation relies on the radio propagation interferences, due to severe multi-path, low probability of a Line-of-sight (LOS) path, reflecting surfaces, and environment dynamic characteristics, such as building restructuring and moving objects. There are three main positioning techniques: trilateration, fingerprinting and proximity.

4.1.1.1 Trilateration

Trilateration is one of the most used techniques by which the location of devices can be determined. The process consists on determining radial distance, obtained by the received signal, from three or more different points. Trilateration can be used on most RF based technologies by calculating distances from two different points. If the position of three access points A, B and C and the distances of MA, MB and MC are known (Figure 4.1), it is possible to obtain the M relative position by the trilateration method.

Several methods can be used to measure the distance from a receiver to a transmitter unit, such as Time of Arrival (TOA), Time Difference of Arrival (TDOA), and Received Signal Strength (RSS). The TOA technique, also known as Time of Flight (TOF), assumes that the distance between the mobile device and the base station is directly proportional to the propagation time. In TOA, only the one-way propagation time is measured, and the location is calculated according to the elapsed time. The major drawback of this technique is that all transmitters and receivers must be precisely synchronised. Moreover, transmitting signals must carry a timestamp value in order to the measuring unit discern the amount of elapsed time.

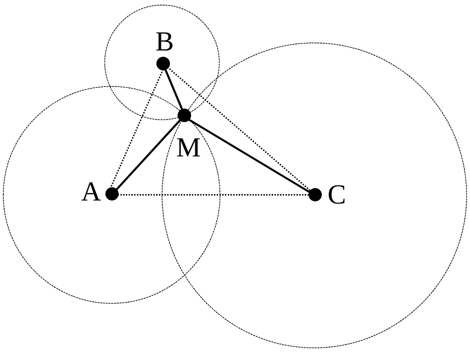


Figure 4.1: Trilateration Concept

TDOA determines the relative position of mobile devices by examining the time difference at which the signal arrives at multiple devices, instead of an absolute time like TOA. The mobile devices analysis the different arrival times to the base stations to calculate its distance. Using TDOA method, only the base stations need to share a precise time reference, not imposing any requirement on the mobile device.

Both TOA and TDOA have several drawbacks. Particularly, for indoor environments it is unlikely to find a LOS between transmitter and receiver, which severely affects propagation time, thus decreasing the accuracy of the estimated location. An alternative approach is to estimate the distance using the attenuation of the emitted signal. Thus, measuring the difference between the emitted and received signals, which is translated into a RSS value, it is possible to estimate the location of devices.

4.1.1.2 Fingerprinting

In contrast with Trilateration, RF based fingerprinting algorithms first collect features (fingerprints) of a scene and then estimate the location of devices, by matching (or partially matching) real time (online) measurements with fingerprints. Most of these algorithms define location fingerprints based on RSS values, previously obtained (offline). Thus, the fingerprinting technique must occur in two stages: the offline gathering of fingerprints, where multiple measurements of known locations are stored in a database, and a online location estimation, which obtains the most suitable match from the database. The major challenge of this technique is the dynamic environments, since building layouts and arrangement of objects are likely to change, thus affecting RSS measurements. Most Location Fingerprinting (LF) based positioning algorithms rely on probabilistic methods or K-nearest-neighbor (Knn) for pattern recognition.

4.1.1.3 Proximity

Proximity algorithms determine symbolic locations. Typically, it relies on the installed base stations, each classified to be in a known position. When a mobile device is detected by the BS antenna, it is considered to be located in its coverage radius. Moreover, when multiple antennas detect a device (overlapping), is is considered to be located in the BS with the strongest signal, whereas the RSS value is typically used. This technique is simple to implement and it offers reasonable results in Radio Frequency Identification (RFID) and IR based algorithms, due to their low range.

4.1.2 Systems and Solutions

There are many proposed wireless location solutions, using different technologies, scopes and with different accuracies. Two main approaches are used, when developing a location system. The first contemplates the usage of existing infrastructures, such as access points, and used them to locate devices. The second approach consists in the development of the entire system, the network infrastructure and the signalling system. In the first, installation costs are low and usually, the system can be deployed on different scenarios, without additional costs. On the other hand, the development of a new system, only designed for location is favourable, potentially increasing the positioning accuracy. In this document, pioneer and recent solutions are discussed, regarding their technology, accuracy, range and dimension capabilities.

4.1.2.1 Infrared

Infrared (IR) based technology provides advantages such as the restriction of signals within rooms (IR beams do not cross opaque obstacles) and immunity to Electromagnetic Interference (EMI), in contrast with RF based technologies. Furthermore, the signal power can be adjusted in order to cover only small areas.

The Active Badge [Want et al., 1992] system was a pioneer contribution in location sensing systems and source of inspiration to many following projects. The main goal of this solution is the ability to locate persons or objects inside public buildings like hospitals. Each person wears a badge, which emits an IR signal within every 10 seconds. The sensors placed at known positions are responsible to receive the unique identifiers and relay these to the location manager software. Emitted signals are reflected by surrounding materials and therefore are not directional when used inside small rooms. Pfeifer *et al.* [Pfeifer and Elias] proposed an hybrid IR/RF solution suitable for outdoor and indoor location tracking. Each user wears a IR receiving and RF transmitting badge. The scenario is equipped with stationary, standalone infrared ID beacons and with stationary RF receivers with LAN (or WLAN) interface. In contrast with Active Badge system, IR transmitters are scattered through the scenario to reduce the cost of badges, as receivers are around 10 times cheaper. Upon receiving a new IR ID beacon, badges transmit a message to the base station via RF, informing the management system of its location. At outdoors, the sunlight interferes with the IR transmission. In this case, the badge is localised relatively, by proximity measurements using the Receiver Signal Strength Indication (RSSI), to the base station.

4.1.2.2 WLAN (IEEE 802.11)

The WLAN standard has become widely spread across public hotspots, corporative locations and many types of mobile devices. The dominant role of this technology appeals to the utilisation of existing infrastructures for indoor and outdoor location sensing. The typical accuracy of WLAN positioning systems using RSS is approximately 3 to 30 meters.

Bahl *et al.* [Bahl and Padmanabhan, 2000] proposed an WLAN indoor location tracking system called RADAR. In this work, two main types of approaches are employed to determine user location: empirical model and radio propagation model. The first depends on a database that consists of previously measured signal strength of points, recording user orientation and signal strength for each Base Station (BS). In the second approach, authors adopted the Floor Attenuation Factor (FAF) and Wall Attenuation Factor (WAF) models [Seidel and Rappaport, 1992], taking into consideration the number of obstructions walls and material types between the user and the BS. These values depend on the building layout and must be derived empirically. The accuracy of RADAR is approximately 2 to 3 meters. Despite the low installation cost, RADAR has a big disadvantage. The collection of reference data is always necessary for both approaches. Each change in a room structure requires an update of the reference model. Sanchez *et al.* [Sanchez et al., 2009] presented a location determination method which limits search area to a starting point, inferred by typical trilateration. Upon the trilateration of a device to a point, search will be constrained to a surrounding area (search are reduction), thus decreasing computation cost. Results show an acceptable accuracy where 90% of the cases is within three meters. Hossain *et al.* [Hassan et al., 2010] proposed a 3D location tracking system, using the trilateration technique based on RSSI values, as well. The testbed scenario consisted on an area with 12 meters x 12 meters of size with two floors, whereas three Access Points (APs) were attached at fixed positions. During tests, mobile nodes moved freely, measuring RSSI values and sending them to the central processing server, which is responsible to calculate locations.

4.1.2.3 Bluetooth (IEEE 802.15)

Compared to WLAN, the Bluetooth gross bit rate is lower (around 2 Mbps max.), and the range shorter (typically 10 - 15 meters). On the other hand, Bluetooth technology is lighter standard, providing a less complex stack and support to other networking services in addition to IP.

Rodriguez *et al.* [Rodriguez et al., 2005] proposed a new indoor location system based on the Bluetooth technology. Access points of the network are used to provide network access and location estimates. Nodes measure the RSS values received from access points and sent them to a central server, through the network, where location calculations are performed. Since processing power is concentrated in a powerful/wired up machine, it is possible to use any kind of location estimation algorithm. Furthermore, the location system uses a reference scene analysis of signal strength, previously performed. This approach is similar to RADAR

[Bahl and Padmanabhan, 2000], where RSSI readings are sent from mobile nodes to the server in tuples.

Raghavan *et al.* [Raghavan et al., 2010] proposed a location system, for indoor environments, suitable to any technology that provides RSSI values, such as Bluetooth and WLAN. However, since it is designed to locate robots, the authors chose to use Bluetooth, as power consumption is significantly lower than WLAN, despite of providing a higher data rate. This approach uses a different trilateration method, namely iterative trilateration, as employed in [Lau and Chung, 2007]. The method can provide more accurate results, however at a higher processing cost, by discarding the points with a low error, and repeating the computation process to the remaining.

More recently, Cruz *et al.* proposed a 3D indoor location and navigation system using Bluetooth radio technology [Cruz et al., 2011], implemented using Java and J2ME. Location calculation is performed using the K*nn* (k-Nearest Neighbor) [Jiangsheng, 2002] method and its conducted by the mobile device.

4.1.2.4 Radio-frequency Identification

Radio-frequency Identification (RFID) is a technology capable of storing and transmitting data to an RF compatible circuit. Typically, these systems are composed by RFID readers and RFID tags. The RFID reader is able to read the data emitted from RFID tags, using a defined RF and protocol exchange information. Moreover, RFID tags can be either passive or active. Passive RFID tags operate without a battery and require stronger signals from the reader. They are much lighter, smaller and less expensive than active tags. Basically, passive tags reflect the RF signal transmitted to them from the reader and add information by modulation the reflected signal. The signal strength returned from these tags is, however, constrained to very low levels (readers power the tags), ranging from 1 up to 3 meters. Active RFID tags are small transceivers, which actively broadcast data, such as their ID, in response to an interrogation. The advantages are clearly the transmission range and the low required signal strength emitted by readers.

LANDMARC [Ni et al., 2003] is an indoor location sensing system using active RFID, aiming to locate objects. The infrastructure consists of RFID readers, active RFID tags and a management server. All objects must be tagged with an active tag. Active tags are also deployed across the scenario, acting as reference tags, aiding the location process with a low installation cost. This approach requires signal strength information from each tag to readers, whereas location estimation is performed using the K*nn* (k-Nearest Neighbor) [Jiangsheng, 2002] method. The server communicates with the readers to receive RSS measurements and calculates the estimated positions of targets. The main disadvantage of this approach resides in the sequential scan of all reading ranges, which takes about one minute per cycle.

Cheng *et al.* proposed a system to improve localisation accuracy of objects in hospitals for health and safety monitoring, namely COMPASS [Cheng et al., 2009]. The innovation of the COMPASS algorithm is the ability to estimate the position of tracking tags based on a cluster (community) of four reference tags. Rather than trying to compute the position of the tracking tag based on each individual reference tag, a community based localisation method is employed to improve tracking accuracy. This approach has more accuracy and lower error when compared to LANDMARC results. However, a larger quantity of RFID readers, acting as reference tags, are needed.

4.1.2.5 Zigbee (IEEE 802.15.4)

ZigBee is an RF wireless technology designed for small devices, including several benefits such as low power consumption, simplicity of the stack, and easy deployment on Wireless Sensor Network (WSN).

Cheng [Cheng, 2009] proposed a room-based location technology using ZigBee wireless technology. Two ZigBee nodes are placed inside each room, one at the door, with the antenna pointing inwards the room and adjusted within 1.5 meters, and a second in a unspecific wall, adjusted within 10 meters. When the user tag passes the door or room and the secondary node senses the user tag, it can be certain that the user is in that specific room. The node positioned at the door is crucial as it avoids location miscalculations when the range of secondary nodes overlap. This approach is quite different from the majority of proposals in literature, since it does not need to calculate location based on RSS information, therefore requiring a very low processing power.

Bras *et al.* [Bras et al., 2010] proposed a ZigBee location protocol based on WSN. The main objective of this approach is to reduce power consumption. To achieve that, authors constrained the traffic exchanged by the coordinator and used a custom built routing protocol. The coordinator is the core processing module of the network, which receives the RSSI values from mobile nodes in the ZigBee network and calculates its relative location. Furthermore, the routing protocol provides two modes of location: HiRSSI mode is a macro-based location, without need for calibration, ready to use from the start up. However, if more precise location is needed, multi-RSSI mode supports more complex algorithms based on RSSI analysis, collecting RSSI values from several devices.

4.1.2.6 Global Positioning System

The Global Positioning System (GPS) is one of the most successful position systems for outdoor environments. Thus, the discussion of this system alone, is somehow futile. There are, however, hybrid systems than combine GPS with other technologies, capable of providing indoor and outdoor location sensing.

Misra *et al.* pioneered in an hybrid GPS solution, called Assisted-GPS (A-GPS), providing outdoor and indoor positioning with an accuracy of 5-50 meters. A-GPS technology uses a server with a reference GPS receiver to improve accuracy when only weak GPS signals are available. The references from the database can be obtained via internet connection, collecting the necessary information according to the wireless mobile network.

Guillemette *et al.* [Guillemette et al., 2008] proposed an hybrid RFID-GPS location system, applied to a real scenario, whereas indoor and outdoor positioning is performed with the RFID and GPS technologies, respectively. The system was designed to monitor security guards patrolling a campus. For outdoor tracking, GPS receivers were integrated with radios, which periodically report its position using a defined RF. In indoors, RFID tags were installed in buildings, able to receive beacons sent by the RFID beacons, also installed on the radios. When RFID beacons receive responses to the emitted signals, a message is sent via WLAN technology to the WLAN base stations, also installed in the campus. With this system, it is possible to determine, in real time, the location of guards. Authors do not discuss energy concerns, however, since each radio is equipped with a WLAN card, a GPS receiver and an active RFID, the devices are likely to have low autonomy.

4.1.2.7 Ultrasound

The most important property of ultrasonic location systems is that they have the capability to be fine-grained, meaning that it is possible to estimate location with a high degree of accuracy. This occurs because the speed of ultrasonic waves in the air is sufficiently slow to allow the TOF of a signal to be accurately measured between sender and receiver.

The Bat system [Harter et al., 2002] presents a location approach based on ultrasound. Each person or object carries a device called Bat that periodically sends an ultrasonic signal. Receivers are placed to fixed positions at the ceiling of rooms, and connected to a wireless network. Analysing the arriving times, provided by several receiving units, the core management system calculates the position of devices. This project shows that ultrasound provides an high accuracy location sensing, however ultrasound is highly vulnerable to interferences.

Cricket [Smith et al., 2004] is another ultrasound based location system. In contrast with the Bat system, mobile devices are responsible to determine the location by themselves, ensuring privacy to the users. Also, instead of receivers, beacons are placed in the ceiling, and periodically send radio and ultrasonic signals. Using multiple signals from different beacons, the mobile device calculates the current position.

4.1.3 Summary

This subsection discussed some of the most important positioning techniques and solutions. Different technologies were presented, exhibiting the main trade-offs between them. Table 4.1 shows the analysed location sensing solutions. It was observed that the main challenge of location sensing lies in indoor environments, since GPS technology is capable of providing fair results outdoors. Different technologies imply a different granularity and accuracy of the location system. Thus, some proposals are hybrid, compiling multiple technologies. However, these solutions have energy issues, due to the necessity of multiple hardware components to support the employed technologies.

Table 4.1: Wireless-based Location Sensing Solutions

Solution	**Technology**	**Accuracy**	**Indoor range**	**Dimension**
Active Badge	IR	Exact room	< 6m	2D
LocSens	RF	1.05m - 2.90m	< 20m	2D
COMPASS	RFID	-	-	2D
RADAR	WLAN	2m - 3m	25m - 50m	2D
HLPS	IR + RF	Exact room (indoor)	< 8m	2D
LANDMARC	RFID	1m	50m	2D
ZRL	ZigBee	Exact room	10m	2D
Bat	Ultrasound	0.03m	6m	2D
Cricket	Ultrasound + RF	0.06m - 0.15m	6m	2D
3DIL	Bluetooth	2m	6m - 10m	2D/3D
ILB	Bluetooth	2m - 4m	6m	2D
AMRL	Bluetooth	0.198m - 0.656m	10m	2D
WLD	WLAN	3m - 4m	50m	2D
ILT	WLAN	1m - 4m	25m	3D
A-GPS	GPS	5m - 50m	NA	3D
RTLS	GPS + RFID	-	-	2D/3D

Location sensing solutions are complex, particularly the trilateration and fingerprinting methods. The trilateration method requires a significant amount of propagation information to be continuously exchanged between nodes in order to provide updated position informations. In addition, nodes must, also continuously, compute their position based on the received information. This process results in a high network overhead and processing time, diminishing the limited resources of nodes. Fingerprinting is also not an optimal solution to wireless ad hoc networks, particularly for two reasons. First, one or more *central* nodes must contain the fingerprint database and receive the fingerprint data of all nodes in the network. This results in a large overhead and requires *central* nodes to be always reachable, which is very complex in a mobile ad hoc network. The second problem is that a fingerprint database of the scenario must previously exist, limiting the intrinsic capabilities of MANETs, which are usually spontaneously deployed in unknown scenarios.

4.2 Clustering for Indoor and Dense Networks

There is a large variety of clustering schemes in literature, with different mechanisms and objectives, aiming to build a suitable hierarchical structure in order to provide an efficient routing in MANETs. Despite the goal of the majority of schemes, which aims at the impromptu deployment of wireless networks in remote environments, clustering can also be an asset in common scenarios, where network infrastructures are present. Typical Wireless Local Area Networks (WLANs) infrastructures do not efficiently support a large quantity of associated nodes, becoming overloaded and consequently unresponsive. A high number of users associated with an Access Point (AP) frequently results in a poor network performance. Increasing the number of APs is often not the solution as the levels of radio interference also increases [Zhang et al., 2015]. A possible solution to address this issue is the utilisation of ad hoc networks under these high density network environments.

The Clustering for Indoor and Dense Networks (CIDNET) is a distributed clustering scheme designed for dense cooperative environments, where existent network infrastructures are insufficient. This clustering solution is designed for ad-hoc networks, utilising the surrounding WLAN infrastructure as context information to improve cluster management. APs are used as proximity location references, in order to facilitate cluster creation and management.

As described in Section 4.1.3, the trilateration and fingerprinting location methods are complex to employ in MANETs. Thus, CIDNET is based on proximity location, relying on APs to determine the location information for the entire network. Nodes scan for AP broadcasts and create clusters according to their identification. This cluster creation mechanism is more efficient than the one employed in SALSA since all nodes in communication range with an AP are immediately assigned to a cluster, not requiring an initial waiting period for cluster creation. Nonetheless, CIDNET implements several mechanisms of the SALSA scheme, namely the automatic clustering balancing and the determination of most suitable joining cluster.

4.2.1 Location Sensing and Dissemination

CIDNET introduces a novel node type called anchor node. Upon network deployment, nodes are typically scattered across the entire network scenario. In a WLAN infrastructured environment, some nodes will be located in communication range with some APs, others will not have this connectivity. The nodes capable of receiving broadcasts from at least one AP assume the anchor node role. The remaining nodes are simply called blind nodes.

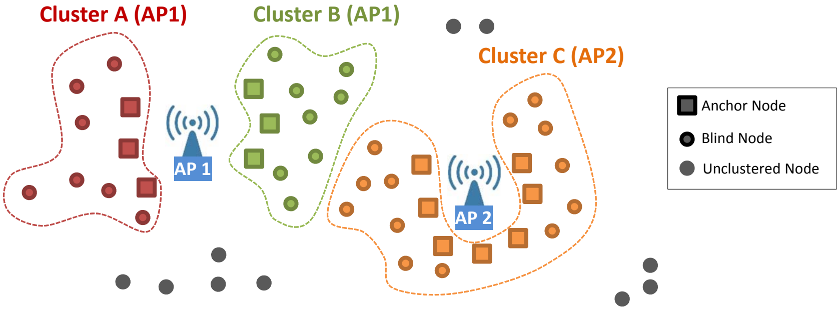


Figure 4.2: CIDNET Clustering Example

Anchor nodes are responsible of creating clusters based on the received Basic Service Set Identification (BSSID) from APs. Should an anchor node receive multiple BSSIDs it chooses the one with the highest Received Signal Strength Indication (RSSI) value as proximity location reference. After choosing the reference BSSID a cluster is created with a globally unique ID and a `PING` message is broadcasted to announce the existence of a cluster. Neighbour blind nodes, upon receiving the message decide whether to join the cluster associated; and forward the received broadcast to its neighbours. Broadcasts are repeatedly forwarded until a configurable Time to Live (TTL) reaches 0. Figure 4.2 shows a possible clustering scenario of CIDNET. As depicted, only anchor nodes have connectivity to APs. Blind nodes receive the broadcasts of the created clusters and join the clusters.

To be noted that multiple clusters can be associated to the same AP, like Cluster A and Cluster B. This occurs when two or more clusters do not have connectivity. In the best case scenario, each broadcast would be associated with one single cluster, however this situation is not always possible. Looking at Figure 4.2, Cluster A and B are associated with one specific AP broadcast and they cannot be merged into one single cluster, since their nodes do not have connectivity. Additionally, there are nodes that remain completely isolated and cannot be associated to the cluster structure. This situation occurs when nodes do not have any connectivity, even through neighbour nodes, to APs or cluster broadcast messages did not reach them due to TTL expiry.

4.2.2 Node States

Nodes can be in one of three distinct states, namely *Unclustered*, *Clustered* and *Clustered-GW*. The *Unclustered* state typically represents a temporary role, as the node is waiting to be assigned to a cluster. In this state, when the node discovers at least one AP it becomes an anchor node with the purpose of creating a new cluster. However, in the meantime if the node receives `PING` broadcasts indicating the existence of clusters, it assumes the blind node role and decides, based on the received broadcasts, what is the best cluster to join using the same mechanism of SALSA. The *Unclustered* state occurs on two different situations:

1. Node isolation (geographic position) - in this case the node does not have any in-range neighbour nodes or AP's, therefore cannot create or be assigned to a cluster
2. Node isolation (TTL expiry) - in this case, nodes have connectivity to neighbour nodes that may be clustered, however due to the TTL expiry of AP broadcasts, they cannot be associated with a cluster. This mechanism is necessary as it prevents the creation of very large clusters, leading to a higher instability of the network

Nodes in the *Clustered* state usually represent the majority of nodes in the network, either anchor or blind nodes, whereas all neighbour nodes must belong to the same cluster. Therefore, the communication with foreign nodes (i.e. nodes assigned to a different cluster) is performed through gateway nodes.

Finally, the *Clustered-GW* state is assigned to nodes that have connectivity with its clustermembers and with foreign nodes. Thus, they are responsible of forwarding inter-cluster maintenance messages and typically are located on the edge of clusters.

4.2.3 State Transitions

In a MANET the majority of node state transitions is related with mobility, however other situations can occur.

Unclustered* to *Clustered This transition occurs when a node becomes aware of at least a cluster or it has direct connectivity with an AP. In the first situation, the node has to evaluate which is the best cluster to join, employing the best cluster metric of SALSA, based on the received `PING` broadcasts. In the second situation, upon receiving an BSSID from an AP, it automatically creates a new cluster and broadcasts a `PING` message to announce its neighbours the presence of a cluster.

Unclustered* to *Clustered-GW This transition is very similar to the previous, but with one difference. When a node becomes clustered, it is considered a gateway if it has direct connectivity with neighbour nodes belonging to different clusters.

Clustered* to *Clustered-GW This transition occurs when a node becomes aware of clusters, excluding its own.

Clustered-GW* to *Clustered Whenever a clustered gateway node loses connection with all its foreign clusters, it automatically transits to a normal clustered state.

Clustered/Clustered-GW* to *Unclustered A node becomes unclustered when willingly disconnects from the network or loses connection with all its neighbour nodes. When this situation occurs, it is necessary to verify the consistency of the cluster, i.e. guarantee that all clustermember nodes can communicate with each other.

4.2.4 Summary

This section described CIDNET, a clustering scheme aiming to improve the stability of dense networks in order to provide a reliable cooperative environment. This scheme uses existent WLAN infrastructures as location references to create evenly distributed clusters in order to improve the stability and management of the cluster structure.

4.3 Adaptive and Location-aware Clustering

Location awareness is a key feature in distributed networks, particularly for clustering schemes. With location sensing mechanisms, each node can join and change clusters more efficiently, with less overhead. Since nodes are assigned to clusters more rapidly, the amount of time within clusters will increase, therefore improving the stability of the network.

Section 4.2 described CIDNET, a scheme designed to operate in dense indoor scenarios using the existing WLAN infrastructures as reference points for cluster creation. In this scheme, nodes that are not in communication range to created clusters around APs cannot be assigned to the cluster structure. Attempting to solve this problem, this section introduces DiLoC, a scheme based on indoor location, designed to operate in infra-structureless as well as infrastructure network environments.

DiLoC is based on proximity location, relying on devices scattered along the network to determine location information for the entire cluster structure. Generally, there are two distinct types of methodology in order to obtain proximity location. In the first methodology it is assumed that the deployment scenario does not have network infrastructures, therefore it is necessary to pre-elect special nodes to provide location awareness. The second methodology takes advantage of the existing WLAN network infrastructure, utilising its Access Points (APs) as references to ease the construction and maintenance of clusters.

The DiLoC scheme contains three different approaches regarding the target deployment scenario: No Infrastructures (NI), Basic Infrastructures (BI) and Advanced Infrastructures (AI). In the first approach the algorithm does not rely on any network infrastructures to retrieve location information. The second approach only relies on the existing network infrastructures. Finally, the third approach is designed to operate in hybrid scenarios, where network infrastructures are not certain to cover all areas.

4.3.1 General Algorithm Description

The main purpose of DiLoC is to build a low overhead clustering structure, aiming to increase network stability with node location awareness. It is designed to build a cluster topology in a distributed fashion (contrary to the clusterhead paradigm), providing a light hierarchical structure for routing. DiLoC is a new designed clustering scheme, which implements some of SALSA features, such as the load-balancing mechanism, the packet structure and the evaluation of optimal cluster to join. The load-balancing mechanism, which acts progressively during execution, distributes nodes across clusters based on their current size. Before the maximum capacity of a cluster is reached, nodes are assigned to a neighbour cluster or, in cases this operation is not possible, builds new clusters to receive excess nodes, keeping an even topology.

4.3.1.1 Node States

Nodes can be in one of three distinct states, namely *Unclustered*, *Clustered* and *Clustered-GW*. The current state of a node strongly depends on having connectivity with its neighbour nodes. Two nodes are considered to be in communication range to another when they are able to exchange information. In DiLoC, a *Ping* message is used as a beacon to determine if two nodes are in-range.

The *Unclustered* state typically represents a temporary role, as the node is waiting to be assigned to a cluster. In this state, when the node discovers neighbours, it waits a predefined period of time in order to calculate the best candidate cluster to join.

Nodes in the *Clustered* state usually represent the majority of nodes on the network, whereas all in-range nodes must belong to its cluster. Thus, the communication with foreign nodes (i.e. nodes assigned to a different cluster) is performed through gateway nodes.

Finally, the *Clustered-GW* state is assigned to nodes that have in-range foreign nodes, i.e. they must have direct connectivity with at least one different cluster. Thus, they are responsible of forwarding inter-cluster maintenance messages and typically are located on the edge of clusters.

4.3.1.2 State Transitions

The *Unclustered* state occurs on two different situations:

1. Node isolation - in this case the node does not have any in-range neighbour nodes, therefore cannot create or be assigned to a cluster
2. Cluster transition - the management of clusters occasionally requires nodes to change clusters, due to cluster balancing. In this phase, nodes can be unassigned from a cluster.

Unclustered* to *Clustered This state occurs when a node becomes aware of an in-range cluster or an unclustered node. In the first situation, the node joins the cluster automatically. However, if the node only detects unclustered nodes, a new cluster is created to adopt the unclustered nodes.

Unclustered* to *Clustered-GW This transition is similar to the previous, but more than one cluster is discovered. Firstly, the node calculates which is the best, taking several parameters into account: number of in-range nodes for each cluster and the size of clusters. The greater the number of in-range nodes, the stronger connection to the cluster. However, if the size of the cluster is high, possibly close to the maximum allowed, this cluster would be a bad choice. To measure this trade-off, a new metric is utilised (5.1), namely the best clustering metric (BC), where BC_i is the metric value for cluster i.

$$BC_i = AP_i + \frac{IRN_i}{C} \tag{4.1}$$

AP_i is defined as the number of the available positions in cluster i until it reaches the maximum allowed, i.e. the difference between the maximum allowed number of nodes per cluster and the current number of assigned nodes. IRN_i is the number of in-range nodes belonging to the cluster. C represents a constant value, allowing IRN_i to be less relevant than AP_i, since the number of available positions in a cluster is typically more important than the number of in-range nodes. Thus, the C value should be chosen according to the target scenario. The clusters in scenarios with a high node density have the tendency to be full, thus the C value must be large to provide more relevance to the available cluster positions. On the other hand, in small density networks, clusters have the tendency to be less populated, and a low C value should be chosen, providing more relevance to nodes in transmission range. The cluster with the higher BC value is chosen by the node.

Clustered* to *Clustered-GW This transition occurs when a node becomes aware of clusters, excluding its own.

Clustered-GW* to *Clustered Whenever a clustered gateway node loses connection with all its foreign clusters, it automatically transits to a normal clustered state.

Clustered/Clustered-GW* to *Unclustered A node becomes unclustered when willingly disconnects from the network or loses connection with all its neighbour nodes. When this situation occurs, it is necessary to verify the consistency of the cluster, i.e. guarantee that all home nodes can communicate with each other.

4.3.2 Description of DiLoC Approaches

As briefly described, DiLoC implements three distinct approaches to handle location proximity. The first assumes that the deployment scenario does not have a WLAN infrastructure and therefore, it uses special nodes serving as reference location points. The second approach is based on an existing WLAN network infrastructure, taking advantage of its APs as reference, in order to provide location sensing. The third is a combination of the previous two approaches, aiming to provide an hybrid solution, suitable to the majority of scenarios.

4.3.2.1 First Approach - No Infrastructures (NI)

This algorithm is designed to operate in indoor environments, where network infrastructures are not available. Therefore, in order to manage clusters based on location information it is necessary to employ the following described technique. During the initial period of network deployment (e.g. network cold start), some configured nodes are assigned to a special role, namely anchor nodes, which will serve as location reference points. Anchor nodes are deployed in a scattered fashion in order to cover as much as area as possible, and remain static for a certain period of time. These nodes are responsible to immediately create clusters and announce them using appropriate messages. This process, accelerates the initial topology setup, in contrast with the traditional exchange of messages employed in solutions designed for MANETs. Once an anchor node detects that regular nodes became assigned to its cluster, it becomes a regular node itself, being able to move and disconnect, i.e. following the exact same rules of regular nodes. Anchor nodes are a complementary tool in order to create clusters more efficiently and with an initial balanced distribution of nodes. Moreover, anchor nodes may be non-existent or disconnected at any moment. If such is the case, regular nodes are also able to create clusters, however in random locations, causing the initial cluster topology to be less evenly distributed.

4.3.2.2 Second Approach - Basic Infrastructures (BI)

In contrast with the previous approach, this algorithm is designed to operate in scenarios where location information can be retrieved using the WLAN network infrastructure.

If an wireless ad-hoc network is to be deployed in a scenario where a WLAN network is present with APs' coverage (e.g. university campus), the overhead associated with the distribution of anchor nodes can be avoided. This approach is capable of recognising APs as relative positions, around which clusters can be created and maintained. The obvious drawback is the strong dependence of an existing infrastructure.

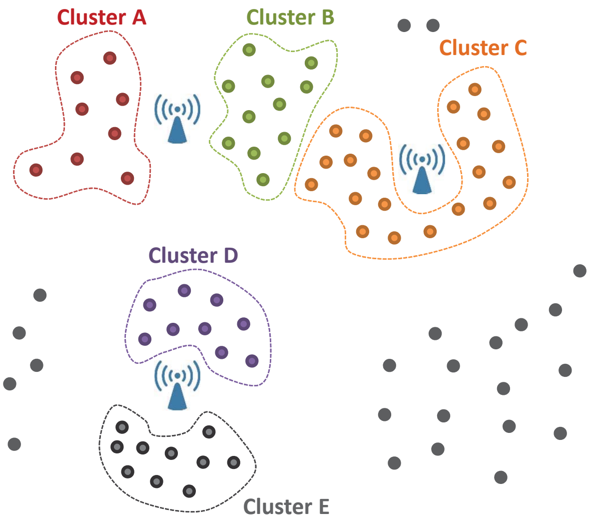


Figure 4.3: Hypothetical Scenario in BI Approach

In the initial network deployment, nodes scan for AP BSSID broadcasts and create clusters according to that information, requiring less message overhead than the previous approach. Nodes around APs are instantly assigned to a cluster according to the BSSID address, whereas the initial waiting period for cluster creation is not necessary. Furthermore, all nodes have the same behaviour, not being necessary to statically configure anchor nodes prior to network deployment.

When an unclustered node detects a new BSSID, it must first analyse if a neighbour cluster exists, i.e. if there are in-range nodes already assigned to a cluster. If a cluster is present, it is immediately assigned to that cluster. Otherwise, it analyses if there are any in-range unclustered nodes. If other nodes are present, it then creates a new cluster identified by a randomly generated globally unique identifier and broadcasts that information, announcing the existence of a new cluster. However, if the node does not have connectivity, it does nothing, remaining unclustered.

In the best case scenario, each broadcast would be associated with one single cluster, however this situation is not always possible. Looking at Figure 4.3, Cluster A and B are associated with one specific AP and they cannot be merged into one single cluster, since its nodes do not have connectivity. The same situation occurs with Cluster D and E.

4.3.2.3 Third Approach - Advanced Infrastructures (AI)

Previous subsections describe two different approaches of the DiLoC algorithm, which contemplates scenarios with and without a network infrastructure. However, a combined approach must be considered, since there are hybrid deployment scenarios, i.e. one part of the scenario has a network infrastructure and the other part does not. Thus, the AI based approach aims to provide a cluster topology covering all areas of the scenario. This approach is a combination of the previous two, with each node operating accordingly to its situation. It is capable of both exploiting network infrastructures as well as creating and managing clusters for isolated nodes (i.e. when nodes are not in-range from the remaining cluster structure). Figure 4.4 shows a hypothetical scenario where APs are deployed. As depicted, clusters are built around APs and isolated nodes are also clustered, in contrast with the BI approach. Since it combines two different paradigms, nodes have to decide which approach to use for different situations. There is a basic rule that defines the behaviour of a node: if it receives broadcasts from an AP or from an existent cluster, it will belong to that cluster. Otherwise (i.e. it is isolated) it will wait for the presence of surrounding clusters, potentially created by anchor nodes. To be noted that should an anchor node be in-range to an AP, the node will immediately resign from that role and become associated with the cluster near the AP.

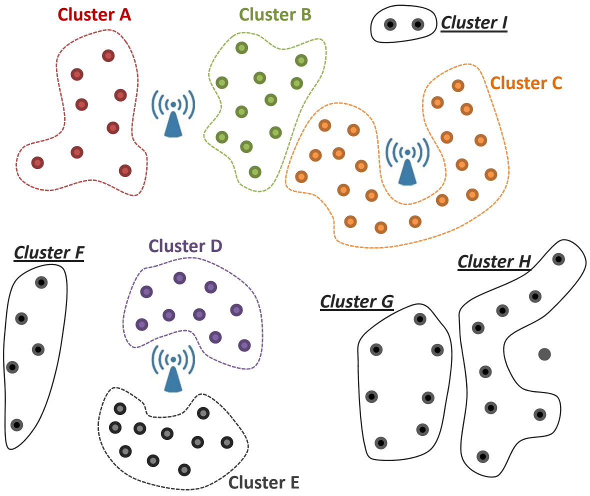


Figure 4.4: Hypothetical Scenario in AI Approach

4.3.3 Algorithm Specification

This subsection illustrates the main structure of the algorithm execution for the unclustered and clustered states of nodes. The three approaches of DiLoC mainly differ in the unclustered state, whereas each follows different rules for cluster creation.

4.3.3.1 *Unclustered* state

In the unclustered state only two types of messages are accepted, namely `PING` and `HELLO`. `PING` messages are used to update the status of nodes and are the only to trigger cluster creation and joining events. `HELLO` messages are only used to update the cluster member nodes table. DiLoC acts differently according to the configured approach. The essential implementation is defined in Algorithms 3, 4 and 5 for the NI, BI and AI approaches, respectively.

Algorithm 3 *Unclustered* - Received Message - NI Approach

```
if received PING then
    update NODE_TABLE
    update CLUSTER_TABLE
    if sender node is clustered then
        start cluster_join_process
    else if this node is anchor then
        start cluster_create_process
        broadcast PING
    end if

else if received HELLO and sender node is clustered then
    update NODE_TABLE
    update CLUSTER_TABLE
end if
```

4.3.3.2 *Clustered* state

In the clustered state, all types of messages are accepted. Once a clustered node receives a *Ping* message from a clustered node, it must check if its *NODE_TABLE* contains the sender node. If it does not, it means that the sender node is a new member of the cluster structure, and a *Hello* message must be sent, in order to provide the information about its home nodes.

The remaining received messages are only accepted if sent by a home node, i.e. a node belonging to the same cluster. If an *Hello* message is received, the maintenance tables are updated and the message is forwarded. If a *Lost* message is received, it means that some node in the cluster lost connection with another. In this case, each receiving node considers all nodes in the cluster as dead, and only are considered alive upon the reception of an *Alive*

Algorithm 4 *Unclustered* - Received Message - BI Approach

```
if received PING then
    update NODE_TABLE
    update CLUSTER_TABLE
    if sender node is clustered and sender_BSSID = this_BSSID then
        start cluster_join_process
    else
        start cluster_create_process
        broadcast PING
    end if

else if received HELLO and sender node is clustered then
    update NODE_TABLE
    update CLUSTER_TABLE
end if
```

Algorithm 5 *Unclustered* - Received Message - AI Approach

```
if received PING then
    update NODE_TABLE
    update CLUSTER_TABLE
    if sender node is clustered then
        start cluster_join_process
    else if (this node is anchor and this_BSSID = NULL) or (this_BSSID != NULL) then
        start cluster_create_process
        broadcast PING
    end if

else if received HELLO and sender node is clustered then
    update NODE_TABLE
    update CLUSTER_TABLE
end if
```

message from each one. Thus, when a *Lost* message is received, all nodes in *NODE_TABLE* are set to dead and a request to check their status is sent, by the broadcast of an *Alive* message. Likewise, upon the reception of an *Alive* message, the sender node is set to alive. This whole process is required for failsafe purposes, since in some cases the lost of a home node is vital for the cluster stability. For instance, if a cluster has the geometric shape of an hourglass, in which only one node is located in the middle, connecting the cluster, the cluster becomes incoherent if it looses connection with that particular node. When a node receives a *Switch* message, the node rapidly changes its cluster to the one specified in the message and the maintenance tables are updated according to the information received. Also, in this case, the message is forwarded for remaining nodes.

Algorithm 6 *Clustered* - Received Message - NI, BI and AI Approaches

```
if received PING then
    update NODE_TABLE
    update CLUSTER_TABLE
    if sender node is new clustered then
        broadcast HELLO

    else if received HELLO and own_cluster_ID = sender_cluster_ID then
        update NODE_TABLE
        update CLUSTER_TABLE
        forward Hello

    else if received LOST HELLO and own_cluster_ID = sender_cluster_ID then
        forward LOST HELLO
        set all dead in NODE_TABLE
        boadcast ALIVE HELLO
        start alive_timer

    else if received ALIVE HELLO then
        forward ALIVE HELLO
        if own_cluster_ID = sender_cluster_ID then
            set all dead in NODE_TABLE
            start alive_timer
        end if
        set sender alive in NODE_TABLE

    else if received SWITCH HELLO and own_cluster_ID = sender_cluster_ID then
        update NODE_TABLE
        update CLUSTER_TABLE
        own_cluster_ID ← sender_new_cluster_ID
        forward Switch Hello
    end if

else if received HELLO and sender node is clustered then
    update NODE_TABLE
    update CLUSTER_TABLE
end if
```

4.4 Evaluation and Results

This section aims to evaluate the performance of CIDNET and DiLoC clustering schemes. Both schemes use proximity location, have the same objectives and are designed for indoor areas. However, CIDNET operates exclusively using WLAN infrastructures in contrast with DiLoC which offers an alternative method of location when WLAN infrastructures are not present. Moreover, both algorithms rely on special nodes, named anchor nodes, which are responsible for cluster creation. In CIDNET, a node is anchor when it is located near an AP, providing an initial reference for cluster formation. DiLoC, on the other hand, uses anchor nodes as temporary reference points for cluster creation, but only when network infrastructures are not present.

To properly examine CIDNET and DiLoC, a simulation evaluation, driven by its main objectives was performed, using the OPNET Modeler [OPNET Technologies Inc. (Bethesda USA), 1986]. The main purpose of this simulation evaluation is to assess the stability and low overhead capabilities of the clustering schemes. Regarding this objective, a set of different simulation environments, featuring the network size and speed of nodes, were defined.

This evaluation compares CIDNET and the three different approaches of DiLoC (No Infrastructures [NI], Basic Infrastructures [BI] and Advanced Infrastructures [AI]) and also the SALSA algorithm, in order to attain the advantages and drawbacks of the different schemes.

4.4.1 Environment and Parameters

The evaluation settings were selected in such a way that they represent, as much as possible, realistic scenarios. In this specification the evaluation parameters are divided into four groups, the fixed-value, the variable-value, CIDNET specific parameters and DiLoC specific parameters, as illustrated in table 4.2.

Given the enormous quantity of different possible scenarios that the combination of parameters provides, only the most significant were chosen. In particular, the parameters that most influence the scalability of the network are the network size (number of nodes) and the maximum speed that nodes can achieve. This simulation study aims to evaluate areas with the existence of network infrastructures, e.g. an university campus. For validation purposes, a random model for mobility, namely the Random Waypoint, was chosen to simulate the movement of people. Also, the average speed on foot of humans does not exceed 2 m/s, which was considered as a good maximum movement speed. Each simulation execution was repeated 30 times, assigning to each a distinct seed value.

4.4.1.1 Specific Requirements - CIDNET

CIDNET relies on APs for the creation and positioning of clusters. Since APs will determine the position of clusters, it would be desirable to evenly distribute them across the deployment scenario. Thus, to evaluate CIDNET, all scenarios will contain 49 APs (7 × 7), placed in a square grid fashion. This CIDNET is not able to assign distant/isolated nodes to the cluster structure, this optimal scenario, with evenly APs was chosen which provides a fair comparison with DiLoC.

Table 4.2: Simulation parameters

Fixed-value parameters	
Simulator	OPNET Modeler 16.0
Field Size (m^2)	500×500
Node mobility algorithm	Random Waypoint Model
Pause time (s)	50
Transmission range (Nodes & APs) (m)	150
WLAN IEEE Standard	802.11b (11 Mbps)
Simulation time (s)	900
Maximum cluster size	50
Ping broadcast interval (s)	1
BSSID broadcast interval (s)	1
Variable-value parameters	
Network size (number of nodes)	80; 160; 240; 320; 400
Node maximum speed (m/s)	0 (static); 2 (dynamic)
CIDNET specific parameters	
Number of Access Points (APs)	49
AP Broadcast TTL	5
DiLoC specific parameters	
Number of anchor nodes	25
Anchor node expiry time (s)	50
Anchor node expiry amount	10
Number of Access Points (APs)	25

4.4.1.2 Specific Requirements - DiLoC

DiLoC also has specific requirements for proper evaluation. As described before, the NI approach requires anchor nodes, which must be configured before network deployment. The BI approach also requires APs to be present in the scenario. Therefore, the number of anchor nodes and APs strongly influences the initial network topology, since clusters will initially be formed around them. A high number of anchor nodes and APs will translate in a high amount of clusters, resulting in fragmented topologies. On the other hand, a low number of anchor nodes and AP will result in clusters with a large number of nodes, which may result in scalability issues. Concerning this trade-off, it is necessary to carefully choose an amount of anchor nodes and APs, suitable with the scenario size and amount of nodes, simultaneously ensuring full scenario coverage.

To evaluate the impact of location information, all scenarios will have 25 anchor nodes, placed in a grid fashion. A lower amount of anchor nodes would result in less clusters and lower initial overhead. However, this amount of anchor nodes was chosen since it is the minimum required amount to provide full coverage for this scenario size. To be noted that anchor nodes do not represent an extra number of nodes in the network. In fact, for a scenario with a network size of 160 nodes, 25 are initially configured as anchor nodes and remaining 135 as regular nodes. Furthermore, anchor nodes become regular nodes after 50 seconds or when the amount of nodes that joined the cluster is greater than 10.

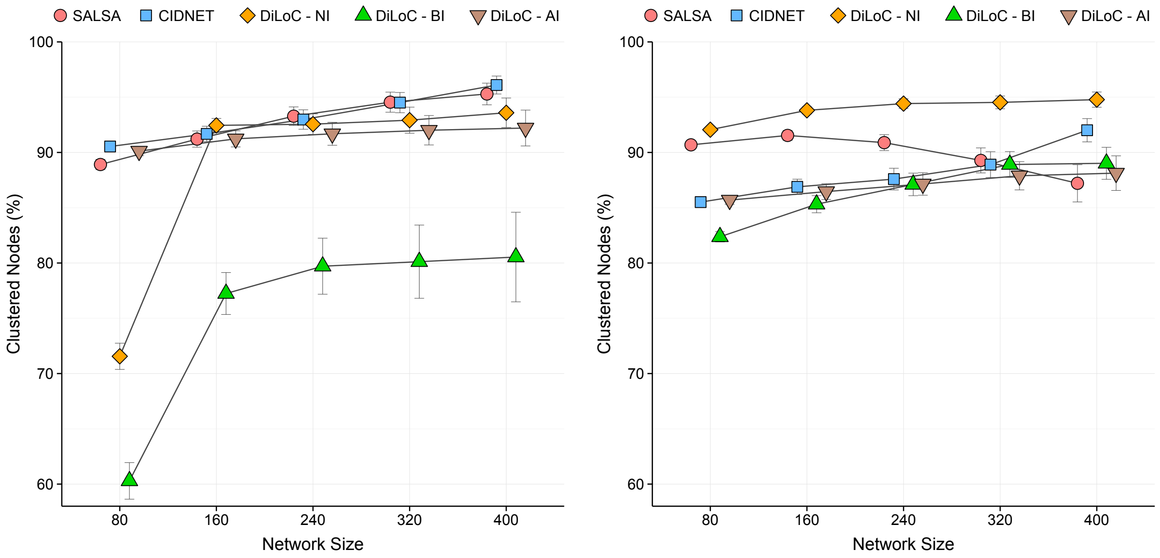
Typically, a lower amount of APs results in less radio interferences, however for a fair comparison, the number of APs present for the BI approach is also 25. APs are also placed in a grid fashion, similarly to the initial position of anchor nodes in the NI approach. Finally, the third approach requires both anchor nodes and APs to be present. For this evaluation, 25 anchor nodes and 25 AP, placed in a grid fashion, were also deployed. Once again, the anchor nodes do not represent an additional number of nodes in the network.

4.4.2 Results

This section presents the obtained results from the simulation. The evaluation accounts three main metrics, which were previously described in Section 3.2.2 and the discussion of the obtained results.

4.4.2.1 Number of Clustered Nodes

Figure 4.5 shows the percentage of the average amount of clustered nodes for the static and dynamic scenarios. In the static scenario there is a significant increase of clustered nodes from network sizes of 80 to 160 in the NI and BI approaches. In the 80 nodes scenario, the connectivity is reduced, due to the low density of nodes, whereas in the 160 nodes scenario, the density is higher, which significantly increases the connectivity and, thus the amount of clustered nodes. This is particularly noticeable in the NI and BI approaches. In NI, clusters are created based on the scattered anchor nodes, however 25 anchor nodes are not able to provide coverage to all nodes. The same occurs in the BI approach, which relies on 25 APs for cluster creation. On the other hand, in the dynamic scenario, even with a low density of nodes, both approaches are able to provide a much superior average amount of clustered nodes due to node mobility. CIDNET also creates clusters based on the position of APs. However, since its simulation scenario was configured to distribute 49 APs in the area, it provides far better coverage, resulting in a higher amount of clustered nodes, particularly in the static scenario.



(a) Static (0 m/s) (b) Dynamic (2 m/s)

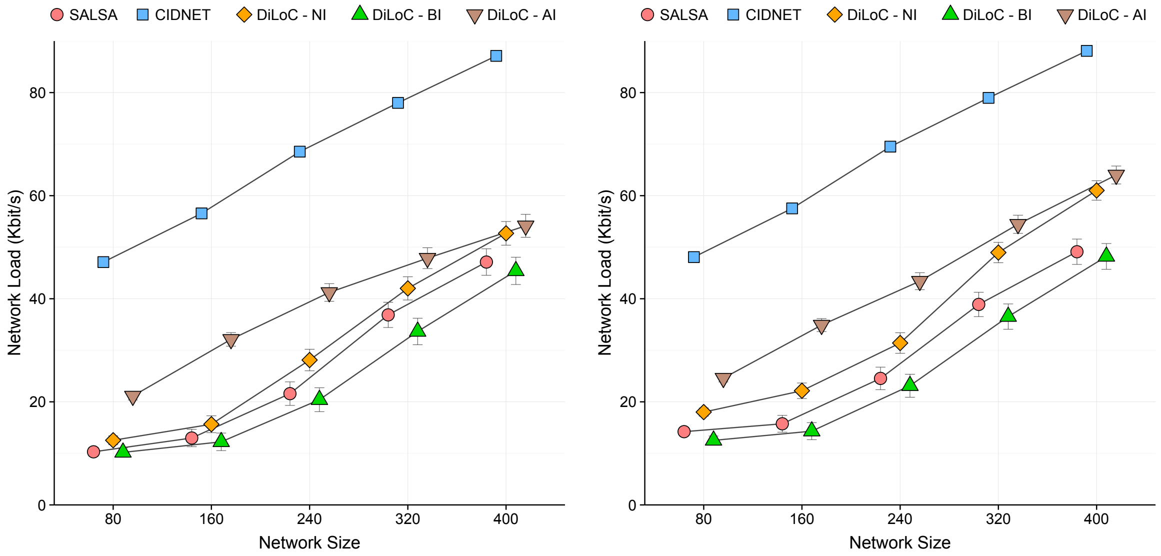
Figure 4.5: Amount of Clustered Nodes (in percentage)

In the static scenario, SALSA shows good results, particularly for largest networks, however it shows a clear scalability problem in the dynamic scenario. This trend was not visible in the first evaluation of SALSA (Section 3.2.2) due to the lower density of the network, when compared to this scenario. The BI approach generally presents the lower number of clustered nodes for both scenarios. Since clusters are only created around AP, there is a great percentage of nodes that are isolated (i.e. not in communication range to clusters), thus affecting the overall amount of clustered nodes. The NI approach presents the best overall amount of clustered nodes. The AI approach also presents very good results for static scenarios, however the amount of clustered nodes drops around 4% in the dynamic scenario, even tough it presents a good scalability trend.

4.4.2.2 Network Load

Figure 4.6 shows the average network load of each scheme. As a global overlook, CIDNET handles clustering with a significant higher overload. This occurs due to the additional broadcasts transmitted by anchor nodes, when in the presence of APs.

As previously described, anchor nodes broadcast messages announcing the creation of clusters, and consequently, blind nodes forward the same broadcasts until a configurable TTL expires. Naturally, if the TTL value is lower, CIDNET will present lower overhead at the cost of assigning fewer nodes to the cluster structure.



(a) Static (0 m/s)

(b) Dynamic (2 m/s)

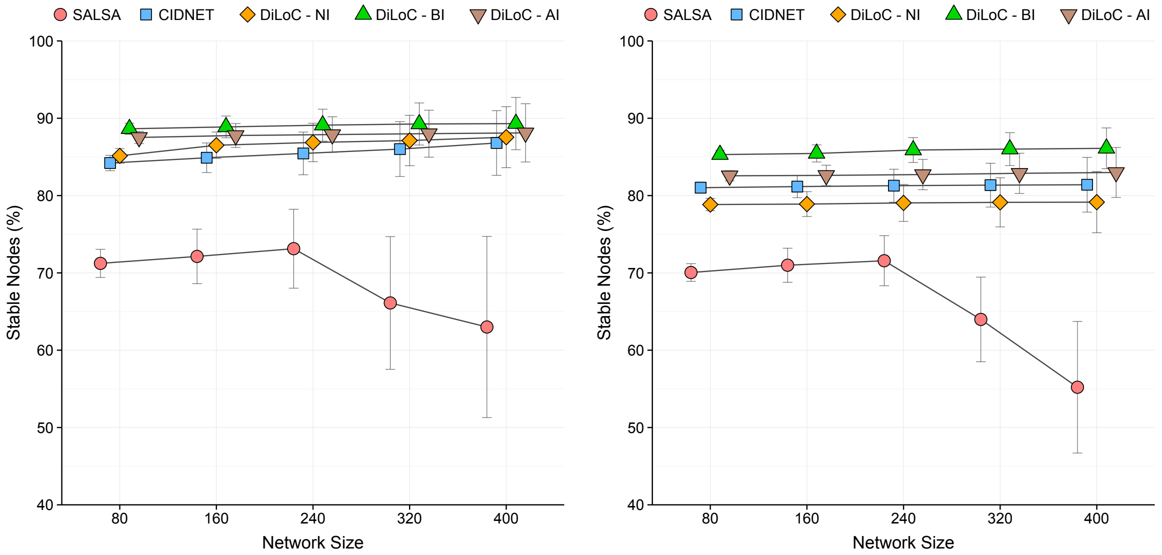
Figure 4.6: Network Load (in kbit/s)

The BI approach and SALSA have a very similar cluster maintenance overhead. The NI approach consumes more bandwidth than SALSA and BI approaches due to the additional communication between anchor nodes and regular nodes, in the initial set up phase. Specifically in the initial set up phase, the NI approach consumes an average of around 14.3% of the total network load in the static scenario and 16.8% in the dynamic scenario. During this phase, anchor nodes broadcast a large amount of *Ping* and *Hello* messages, since they are responsible for the creation of clusters and node table updates of the their cluster.

Finally, the AI approach has a higher overhead than NI, BI and SALSA. The AI approach combines the mechanisms of both NI and BI and additionally each node must decide in which mode to operate (when both modes are available), thus transmitting a larger amount of traffic. During the initial set up phase, AI has a combined average of around 19.1% of the network load in the static scenario and 19.6% in the dynamic scenario.

4.4.2.3 Cluster Stability

As a first analysis of Figure 4.7, the BI approach clearly outperforms the remaining algorithms. Since nodes are created and maintained around APs, this algorithm can deliver a higher amount of stable nodes, even though being one of the schemes with less amount of clustered nodes (Figure 4.5). On the other hand, the NI, BI and AI approaches maintain the percentage amount of stable nodes, whereas SALSA begins to severely decrease in networks larger than 240 nodes, presenting high standard deviations due to the large number of re-clustering operations (i.e. node re-assignments to clusters). Again, this behaviour was not observed in Section 3.2.2 due to the low network density.



(a) Static (0 m/s)

(b) Dynamic (2 m/s)

Figure 4.7: Amount of Stable Nodes (in percentage)

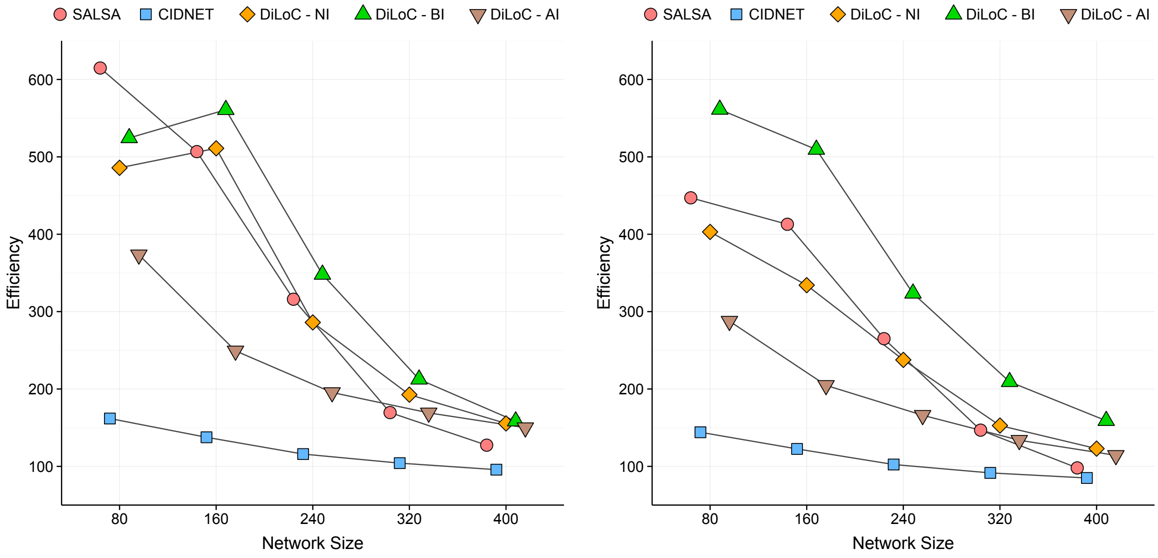
Generally, CIDNET and DiLoC outperform SALSA since they always create clusters around reference points (either APs or anchor nodes), which leads to a more balanced distribution of clusters, leading to a lower amount of re-clustering maintenance operations and thus, a higher stability. Comparing CIDNET to DiLoC, the NI approach and CIDNET present the lowest amount of stable nodes. Given the fact that both algorithms are strictly depended on anchor nodes for cluster creation, the propagation of broadcast messages announcing the presence of cluster consumes time, thus the process of cluster creation is slower which results in nodes being clustered for a smaller period of time. This is more evident in the NI approach for the dynamic scenario, since anchor nodes represent a temporary role and are not referenced to an AP location.

4.4.2.4 Efficiency

This metric assesses the overall efficiency of clustering schemes, combining the previous metrics. The Efficiency is determined according to equation 4.2.

$$E = \frac{CN \times SN}{NetLoad} \tag{4.2}$$

where the CN represents the amount of clustered nodes, SN the amount of stable nodes and $NetLoad$ the network load. Thus, this metric represents a combination of some of the previous evaluated metrics, attempting to represent the efficiency of the clustering schemes. The E value represents the obtained efficiency index value.



(a) Static (0 m/s)

(b) Dynamic (2 m/s)

Figure 4.8: Efficiency

Figure 4.8 shows the obtained efficiency index in static and dynamic scenarios. Since this metric depends on the amount of clustered nodes, it also shows, in static scenarios, an increase of efficiency between the network size of 80 and 160. Despite this fact, SALSA can be more efficient than the remaining schemes for the network size of 80 nodes, in the static scenario. This is due to its high amount of cluster nodes and low network load in such scenario. However, the overall results show that the BI approach is the more efficient scheme and the CIDNET scheme the less efficient, in both static and dynamic scenarios. This occurs because the BI approach has the lowest network load, and the CIDNET approach has the highest network load.

The BI approach does not rely on anchor nodes, taking advantage of the existing infrastructures to provide location information. Thus, the initial overhead caused by anchor nodes does not exist. CIDNET also relies on WLAN infrastructures but through the employment of anchor nodes, significantly increasing the overhead. This concludes that DiLoC is more efficient, when there is a good coverage of WLAN infrastructures. However, such environments may not exist and location information, provided by anchor nodes in AI approach, may be required.

4.4.3 Summary

CIDNET and DiLoC present good performance results, overcoming SALSA in most evaluation metrics. This shows that the location-awareness mechanism is efficient on providing a stabler network topology with a superior number of clustered nodes. Overall results show that the BI approach is more efficient, thus being more suitable than CIDNET to scenarios well covered with network infrastructures. Concerning infrastructure-less scenarios, both the NI approach and CIDNET are suitable, however the NI approach provides a more stable topology in dynamic scenarios. Finally, for hybrid scenarios, the usage of the AI approach instead of the NI approach remains unproven, since in the evaluated scenarios AI presents a significant amount of network load and a relative lower amount of clustered nodes.

4.5 Summary

Location information can be a valuable asset in improving the performance of MANETs. Particularly in clustering, the capability of maintaining clusters based on the location of nodes can significantly increase the stability the clustering scheme. In this chapter, it was presented a brief analysis of relevant location sensing solutions for wireless networks. Moreover, two different clustering schemes were introduced, designed for indoor areas, capable of using existent WLAN infrastructures aiming to improve the performance of cluster maintenance operations. Finally, a performance evaluation was conducted, proving that context location information improves the stability of the cluster topology, with a reduced increase of overhead.

Chapter 5

Social-aware Clustering

Communication networks such as Mobile Ad-hoc Networks (MANETs) involve human interactions and exhibit properties of social networks. Hence, knowledge from social networks can be employed to enhance the performance of communications.

The information gathered from social networks is mostly used to improve the performance of sparse MANETs and Delay Tolerant Networks (DTNs). Sparse MANETs are a class of ad hoc networks where node density is low, and contacts between nodes do not occur very frequently. As a result, message delivery must be delay-tolerant. Traditional MANET routing protocols, such as OLSR, AODV and DSR typically fail to deliver messages since a complete route from source to destination does not exist. To overcome this issue, some studies propose a substantial number of routing protocols, based on social relationships [Hui et al., 2011][Mei et al., 2011][Li et al., 2010]. These protocols exploit social relations among nodes to help making forwarding decisions.

In a regular MANET, whereas the network graph is typically connected, the information of social relations can also be used to potentially improve the performance of the network. Particularly regarding the clustering of a network, social intelligence can be used to improve cluster assignment decisions, thus increasing the reliability and stability of the cluster topology. Inspired by this concept, this chapter proposes the Social-aware Clustering Scheme (SoCS) based on connectivity history between nodes. To the best of our knowledge, this is the first clustering scheme exploring social relationships as a way of improving the performance of clustering management operations. Moreover, social relationships are strongly bounded by social mobility. Thus, to properly study the clustering schemes in wireless ad hoc networks it becomes necessary to use either real movement traces or synthesised traces, generated by social mobility models.

This chapter is organised as follows. Section 5.1 presents a characterisation of social relations and studies some of the most important routing protocols based on social metrics.

Section 5.2 describes the proposed SoCS clustering scheme and Section 5.3 performs its evaluation. Finally, Section 5.4 concludes the chapter.

5.1 Social Properties and Protocols

Social relations among nodes can be used to address the problem of choosing the best relay node in order to forward messages in the best time possible. In this section, a characterisation of social relations is described and a brief study of the most relevant routing protocols is presented.

5.1.1 Social Relations

Social relationships can be utilised to improve the performance of a network. Two stages are required to be able to use social information. The first regards the acquisition of social relations, which can be performed from several sources and methods. In a second stage, the retrieved information needs to be processed and translated into metrics in order to be employed.

5.1.1.1 Acquisition of Social Relations

Real life social relationships can be obtained from multiple sources by self-reported or detected methods.

Self-reported social networks are a type of social relations that are usually collected via a manual insertion of information. Two main sources exist, one is a survey or poll and the other is online social networks.

The survey approach is simple and cheap, however it is limited to the quality of the questionnaire, the willingness of respondents and the motivation of participants to provide accurate answers.

One of the dominant activities in today's internet is online social networks, promoting users to build a virtual community regarding their interests and backgrounds. Social relations in these online social network sites are explicit and can be crawled using software bots, thus providing a clear social network topology.

Detected social networks rely on the analysis of daily activities data, such as mobility traces, call records and location information.

Mobility traces register a variety of encounter information between nodes. These encounter information typically includes contact frequency, contact time, inter-contact time and contact duration. In the real world, acquaintances are more liquefy to encounter and meet for longer periods of time than strangers. Hence, mobility traces are used to identify social relations among nodes.

Similar information is gathered with call records, such as call start time, end time and duration. If two people call each other often there is a large probability that they known each other. Additionally, dynamic social ties can also be detected by analysing the changes in call durations. Some works in literature use call logs to mine social relationships among users [Zhang and Dantu, 2010][Eagle et al., 2009].

Location-based services can provide location information of mobile devices by communication networks, such as Global System for Mobile Communications (GSM) and Long-Term Evolution (LTE), or positioning technologies, such as GPS. By analysing the physical distance between people it is possible to infer their interests and purposes. If some people have similar location information they can be regarded as acquaintances, otherwise they are strangers. This phenomenon is referred as homophily.

5.1.1.2 Characteristics of Social Relations

Due to the complexity of human relationships, its exact characteristics are still unknown. Therefore, it is not possible to directly use social ties information in routing or clustering schemes. Social ties must be analysed according to metrics to identity its features. Some of the most relevant metrics used in literature are as follows.

Degree Centrality determines the number connections of neighbours that a node has (node degree) [Freeman, 1978]. Thus, it can be used to identify the most active nodes in a network. Since degree centrality just considers the information of neighbour nodes, ignoring further nodes, the application scope is limited locally.

Closeness centrality is the mean of geographic distance between a node and all the nodes in the network [Freeman, 1978]. Assuming that all nodes are reachable, closeness centrality is used to estimate how long is necessary to spread information from a given node to all other nodes. However, as this metric is based on a complete and bounded network, it becomes difficult to be determined in a dynamic network.

Betweenness centrality is defined as the amount of shortest paths from all nodes to all nodes that pass through a given node [Freeman, 1978]. It can be used to determine the amount of load of a given node.

Clustering coefficient measures the tendency of nodes to cluster together in a network [Cunha et al., 2014]. There are some variations of this metric. The main used is determined by the number of connections between neighbour nodes divided by the total of possible connections of the node.

Similarity measures common features between nodes, such as interests and locations [Daly and Haahr, 2007]. It is used to identify common groups of nodes.

Selfishness [Li et al., 2010] measures the willingness or cooperation of a node with other nodes. Selfishness is considered a negative effect for message transmission [Zhu et al., 2013], particularly in ad hoc networks. However, it can also be used to reduce traffic in a network with low resources.

5.1.2 Social-based Routing Protocols

Social-based routing protocols exploit social relationships among nodes to make forwarding decisions. Most are usually based on a specific metric, such as centrality, similarity or selfishness. In addition, there are hybrid protocols that consider multiple social metrics to make forwarding decisions.

5.1.2.1 Centrality Based

Centrality is an important characteristic of nodes since it suggests the relative location of nodes in a network. High centrality or popular nodes are more likely to encounter other nodes than unpopular nodes. Centrality-based routing protocols rely on these nodes to forward messages.

PeopleRank is an opportunistic forwarding algorithm that relies on popular nodes (and neighbours of popular nodes) to deliver messages [Mtibaa et al., 2010]. However, using this paradigm often causes popular nodes to become bottlenecks, leading to traffic congestion and fast energy depletion. The contradiction between conserving resources and efficient message transmission is addressed in the Socially-Based Routing for Delay Tolerant Networks (SBR-DTN) [Abdelkader et al., 2010]. This approach measures the connectivity strength between nodes according to the amount of time that elapsed since the last time they met. In addition, this proposal replicates and spreads the same message in the network to increase the probability of reaching its destination, instead of concentrating traffic in shortest paths.

5.1.2.2 Similarity Based

People with similar interests, location or profession are more likely to have social ties. They are also more likely to interact more often than strangers, thus the probability of being located in the same geographic area is higher. The assumption that nodes with higher similarity tend to be in-range more frequently motivated some routing protocols to adopt similarity to make forward decisions [Mei et al., 2011] [Wu and Wang, 2012].

The Social Aware Networking (SANE) routing protocol [Mei et al., 2011] is based on the fact that people with similar interests tend to meet more often than people with different interests. When two nodes meet, they exchange their interest profile and their similarity is computed. The forwarding path of a message is then determined by the similarity of encountered nodes.

Self-reported social networks demand a lower data delivery cost than detected social networks. The protocol proposed in [Bigwood and Henderson, 2008] builds routing tables according to declared social information of users. Messages are simply forwarded by querying routing tables.

5.1.2.3 Selfish Based

Another important social characteristic is selfishness. Selfishness affects willingness of node cooperation in forwarding messages. Most routing protocols are designed on the false assumption that nodes are willing to forward messages for others. Mostly due to energy saving, some selfish nodes are only willing to forward messages to nodes with whom they have social ties, which is harmful for message transmission. There are however, some routing protocols aware of this phenomenon and even take advantage of it to preserve network resources [Zhang and Zhao, 2009a].

The egocentric behaviour of nodes is considered in Give2get [Mei and Stefa, 2012] routing protocol, in which each message is encrypted and anonymous to all intermediate nodes in order to force selfish nodes to cooperate in message forwarding. Give2get uses message replications to maximise the probability of successful delivery.

Although selfish nodes affect the efficiency of message delivery, they can also reduce the number of messages forwarded, which in some cases can be exploited to reduce overhead. In [Zhang and Zhao, 2009b] social selfishness is exploited to reduce the number of message copies and improve delivery speed.

5.1.2.4 Hybrid Approaches

On of the most popular hybrid protocols is SimBet [Daly and Haahr, 2007] which considers multiple social properties to make forwarding decisions. SimBet employs betweenness centrality and similarity of nodes to deliver messages as soon as possible. Similarity is used to measure common characteristics of nodes. Meanwhile, betweenness centrality is used to identify bridge nodes to exchange messages between communities. However, since it is very complex to calculate betweenness of nodes without a complete knowledge of the network, a particular betweenness metric, namely egocentric betweenness, is used to identify bridge nodes with limited local information. Results show that SimBet is capable of a delivery performance close to Epidemic Routing [Vahdat and Becker, 2000] but with significantly reduced overhead. Additionally, SimBet outperforms PRoPHET [Lindgren et al., 2003], particularly when the receiving and sending nodes have low connectivity.

Another successful hybrid protocol is BubbleRap [Hui et al., 2011], which focuses on community grouping and betweenness centrality. In BubbleRap, nodes are structured in groups or communities according to a community detection algorithm. Meanwhile, betweenness centrality is used to identify popular nodes, which have better connectivity. However, since in a dynamic network is difficult to determine betweenness centrality of nodes, a degree centrality limited per a time unit is used instead. In addition, contrary to some protocols, BubbleRap does not spread multiple copies of the same message to increase the probability of successful delivery.

5.1.3 Social Mobility Models

Social mobility models are considered the application of social network theory on the mobility modelling field [Musolesi and Mascolo, 2009]. Mobile nodes are typically carried by humans, hence in most cases, mobility patterns are influenced by human decisions and social behaviours.

The Social Network Theory (SNT) [Musolesi and Mascolo, 2007] mobility model studies social relationships based on social network theory. This model assumes that social ties between individuals are symmetric. Thus, the social relationships of a group of individuals can be measured according to an interaction matrix, which can also be interpreted as the likelihood of geographic colocation of individuals. Synthetic mobility traces are generated based on this matrix, following a model of social attractivity between individuals within a group. In other words, for each individual, the movement is generated driven by attractivity to a given area, which is related with a group location. SNT is validated using real movement traces. The mobility model proposed in [Venkateswaran et al., 2006] is also based on Social Network Theory. Relationships between individuals have weights, which can be asymmetric, i.e. a social tie between two individuals may have two different weights. Thus, paths are generated individually from each individual, biased by the social interaction weights. Additionally, weights are prone to changes according to different locations and time. For instance, if an individual, during daytime is at work, the relationship between him and its co-workers becomes stronger (higher weight). On the other hand, during nighttime the same individual is likely to have stronger ties with his family. The N-Body mobility model [Zhao et al., 2015] is also a social mobility model. However, this work uses a different approach to synthesise mobility traces. Instead of studying social network theory, the model was determined by extracting the characteristics of sample real traces, and synthesise traces that have similar features.

5.2 Social-aware Clustering Scheme

In recent days, with the effortless extraction of social relationships, social awareness can be regarded as an opportunity to improve the performance of clustering management. This section proposes SoCS, based on link history aiming to improve the cluster maintenance operations and provide a reliable cluster structure.

5.2.1 General Description

The main purpose of SoCS is to build a low overhead network topology in order to increase the scalability of the network. Relying on social awareness, nodes are able to maintain a history of connections to neighbour nodes in order to improve maintenance operations, such as aggregation to clusters. SoCS is a distributed clustering scheme (i.e. does not rely on clusterheads) which implements some features of the DiLoC clustering scheme, namely the cluster maintenance messages, the load balancing of clusters and the best clustering metric (BC).

5.2.2 Node States

In SoCS, nodes can be in one of three distinct states, namely *Unclustered*, *Clustered* and *Clustered-GW*.

The *Unclustered* state typically represents a temporary role, as the node is waiting to be assigned to a cluster. In this state, when the node discovers neighbours, it waits a predefined period of time in order to calculate the best candidate cluster to join. The *Unclustered* state occurs on two different occasions:

1. Node isolation - in this case the node does not have any in-range neighbour nodes, therefore cannot create or be assigned to a cluster
2. Cluster transition - the management of clusters occasionally requires nodes to change clusters, due to cluster balancing. In this phase, nodes can be unassigned from a cluster.

Nodes in the *Clustered* state usually represent the majority of nodes on the network, whereas all in-range nodes must belong to its cluster. Thus, the communication with foreign nodes (i.e. nodes assigned to a different cluster) is performed through gateway nodes.

Finally, the *Clustered-GW* state is assigned to nodes that have in-range foreign nodes, i.e. they must have direct connectivity with at least one different cluster. Thus, they are responsible of forwarding inter-cluster maintenance messages and typically are located on the edge of clusters.

The possible node state transitions are defined as follows.

Unclustered* to *Clustered This state occurs when a node becomes aware of an in-range cluster or an unclustered node. In the first situation, the node joins the cluster automatically. However, if the node only detects unclustered nodes, a new cluster is created to adopt the unclustered nodes.

Unclustered* to *Clustered-GW This transition is similar to the previous, but more than one cluster is discovered. In this case, the node determines which cluster is the most suitable, either relying on its link history list or according to a best clustering metric. If a link history between the unclustered node and its in-range clustered nodes exists, this will be the used methodology (further described in Section 5.2.3). However, if no previous connections existed, the node calculates which cluster is the best, taking several parameters into account: number of in-range nodes for each cluster and the size of clusters. The greater the number of in-range nodes, the stronger connection to the cluster. However, if the size of the cluster is high, possibly close to the maximum allowed, this cluster would be a bad choice. To measure this trade-off, the best clustering metric 5.1 is used, where BC_i is the metric value for cluster i.

$$BC_i = AP_i + \frac{IRN_i}{2} \tag{5.1}$$

AP_i is defined as the number of the available positions in cluster i until it reaches the maximum allowed, i.e. the difference between the maximum allowed number of nodes per cluster and the current number of assigned nodes. IRN_i is the number of in-range nodes belonging to the cluster. As a result, the cluster with the higher BC value is chosen by the node.

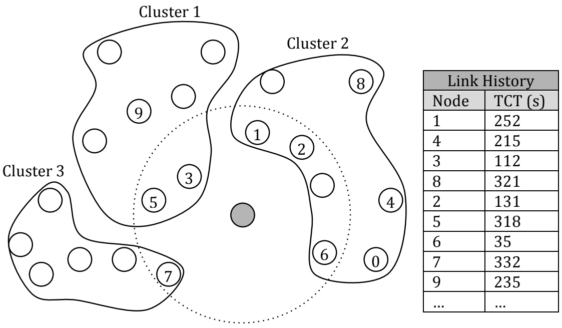
Clustered* to *Clustered-GW This transition occurs when a node becomes aware of clusters, excluding its own.

Clustered-GW* to *Clustered Whenever a clustered gateway node loses connection with all its foreign clusters, it automatically transits to a normal clustered state.

Clustered/Clustered-GW* to *Unclustered A node becomes unclustered when willingly disconnects from the network or loses connection with all its neighbour nodes due to mobility.

5.2.3 Link History

Nodes keep information about their previous connections with neighbour nodes (i.e. at 1-hop distance). This information can be used in future connections to improve both the cluster joining process and duration time of connections. Each node keeps a list of previous neighbour nodes with which a connection occurred. The list contains the total of connectivity time with each neighbour, which is the sum of the amount of time in all connections. Figure 5.1 shows an example of a node making a decision to join a cluster.



Link History	
Node	TCT (s)
1	252
4	215
3	112
8	321
2	131
5	318
6	35
7	332
9	235
...	...

Figure 5.1: Example cluster decision based on link history

Three clusters are in-range of the node, since it has at least one in-range node inside each cluster. Based on the link history list, it can be determined that *Cluster 1* has a total of 665 seconds (sum of nodes 5, 3 and 9), *Cluster 2* has a total of 954 seconds (sum of nodes 1, 2, 8, 4 and 6) and *Cluster 3* has a total of 332 seconds (node 7). Thus, the decision will favour in joining *Cluster 2*. To be noted that, despite nodes 9, 8 and 4 not currently being in-range with the joining node, the latter is still able to see them, since the full table of clustered nodes is broadcasted upon the presence of an unclustered node. Otherwise, in this example, the decision would favour in joining *Cluster 1*.

The utilisation of link history potentially increases the durability of connections, and decreases the complexity of future cluster assignments, particularly with the existence of repeated connection patterns, often observed in social grouping.

5.3 Evaluation of Social-aware Clustering Scheme

The evaluation of SoCS was performed in a simulated environment using the OPNET Modeler [OPNET Technologies Inc. (Bethesda USA), 1986]. The main purpose of this evaluation is to access the overhead and scalability of clustering regarding the gain obtained with social awareness. This study features the performance of clustering, routing and generated traffic.

5.3.1 Environment and Parameters

The performance of clustering and routing protocols in wireless ad hoc networks is extremely sensitive to the evaluation scenario parameters. The evaluation parameters in Table 5.1 were chosen in order to represent a realistic scenario. The objective of this evaluation was to obtain the gain of clustering scalability using social awareness presented in SoCS. The evaluation methodology was defined as follows.

Table 5.1: Simulation parameters

Simulator	OPNET Modeler 17.5.A PL6
Field Size (m^2)	500 × 500
Mobility Model	RWP; SNT
Transmission range (m)	150
WLAN IEEE Standard	802.11b (11 Mbps)
Simulation time (s)	900
Network size	40; 80; 120; 160; 200
Routing Protocol	C-OLSR
Traffic Pattern	
Packet size (*bytes*)	$U(512, 1024)$
Rate (*packets/sec*)	$U(0, 1)$
Number of intra-group packets	5
Number of inter-group packets	5
RWP Parameters	
Node speed (m/s)	$U(0, 2)$
Pause time (s)	50
SNT Parameters	
Node speed (m/s)	$U(0, 2)$
Number of groups (Caveman model)	10

Scheme Comparison SoCS was compared with DiLoC due to the similarity of clustering characteristics, such as distributed cluster topology, type of messages, and maintenance operations.

Routing Protocol In these simulations, the C-OLSR [Ros and Ruiz, 2007a] routing protocol was used. C-OLSR is an extension of the well known proactive OLSR routing protocol, capable of creating an hierarchical network topology, thus supporting any clustering scheme.

Mobility Models As previously discussed, to reach accurate results in a social aware scheme, it must be simulated with social mobility models. Otherwise, the social grouping does not exist. In this evaluation, the Random Waypoint (RWP) and the Social Network Theory (SNT) mobility models are used. With the evaluation using the RWP model it is expected to establish a baseline of comparison, between SoCS and DiLoC schemes. Moreover, in order to assess the gain of SoCS social awareness, the SNT mobility model is also used.

As described in Section 5.1.3, the SNT model generates a matrix of social relations, based on social network theory. After the generation of synthetic traces by SNT, this matrix is fed to each node of the SoCS scheme before execution, in order to build the link history lists. This methodology is used to simulate a previous warm up time of the network. Each index of the matrix is translated to a link history time of nodes. To be noted that during execution time, SoCS updates the link history in each node, according to neighbourhood nodes.

Generated Traffic To evaluate the performance of the network, traffic packets were generated. The traffic pattern is characterized as follows:

- At each time interval $U(0,\ 1)$ 10 nodes are randomly selected to generate packets.
- The first 5 nodes generate one packet each, of size $U(512,\ 1024)$ bytes, to a random destination node, in the entire network.
- The remainder 5 nodes generate one packet each, size $U(512,\ 1024)$ bytes, to a random destination inside its corresponding cluster.

This pattern intends to mimic a real scenario message exchange, with messages traveling within a cluster and across different clusters.

5.3.2 Discussion of Results

This section presents the obtained results from the simulation. The evaluation features the analysis of clustering, routing and traffic performance. Each metric is described further, along with the discussion of the obtained results.

5.3.2.1 Clustering Metrics

The clustering metrics access the performance of SoCS in comparison with DiLoC, using the RWP and the SNT mobility models.

Number of Clustered Nodes This metric provides the number of nodes that are associated with the cluster structure. Nodes that are isolated, i.e. not in-range with any node, cannot be affiliated to a cluster. Thus, scenarios with few nodes are likely to have more unclustered nodes, due to poor connectivity.

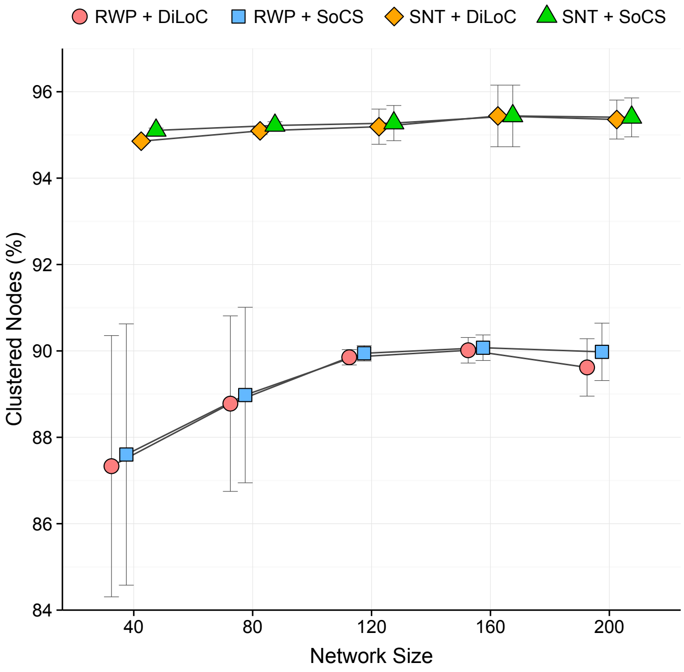


Figure 5.2: Average Amount of Clustered Nodes (in percentage)

Figure 5.2 represents the percentage of average amount of clustered nodes. With the SNT mobility model, the clustering scheme presents around 95% of clustered nodes for all network sizes. With the random waypoint model, the amount of clustered nodes is quite irregular in the low density networks. Moreover, nodes in low density scenarios typically have lower connectivity, hence the lower amount of clustered nodes, particularly with random mobility. It can also be seen that the difference between SoCS and DiLoC is not significative. Thus, the link history of nodes is not significative in this metric, as DiLoC is still capable of clustering as much nodes as SoCS.

Cluster Stability The stability of clusters can be measured according to the amount of time that nodes are affiliated to a cluster, without suffering re-affiliation operations. A cluster stability metric is utilised, which defines a stability time (ST), from which nodes are considered to be stable (5.2).

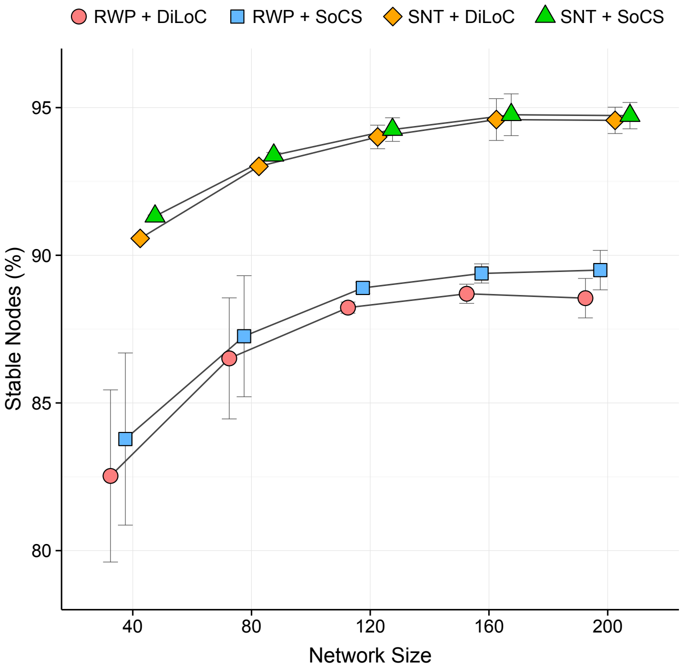


Figure 5.3: Average Amount of Stable Nodes (in percentage)

$$ST = k \times \frac{r \times p}{v \times d} \tag{5.2}$$

where r is the transmission range of nodes, p is the pause time, v the average of node speed (mean value of minimum and maximum speed), d the density of nodes (number of nodes per Km^2) and finally, k represents an arbitrary constant, equal in all simulation executions, enabling the transformation of the ratio to a real execution time. The stability metric provides a measurement on the amount of nodes that were stable during execution, for a period greater than the ST value.

Figure 5.3 shows the average amount of nodes that are stable (within a period greater than ST) in percentage. It can be seen that in both mobility models, the stability of clustered nodes increases for larger networks. In other words, in larger networks, there is a lower percentage of nodes that require re-affiliation operations. As expected, the social mobility provides more stability than random mobility, since nodes remain in the same areas for longer. Once again, in this metric, the increase of stability with SoCS is not very significant, when compared to DiLoC. Particularly with the SNT model, DiLoC is capable of maintaining nodes stable, regardless of social link history.

Clustering Overhead This metric represents the total amount of traffic sent required for cluster maintenance. Figure 5.4 shows the average clustering overhead per second. As expected, with the RWP model there is a significant larger overhead in clustering maintenance. As nodes move randomly the number of re-affiliations is large, forcing the number of required maintenance messages to increase. With the SNT model, however, nodes move with coordination, often in the same area, thus requiring less clustering maintenance messages. Another aspect to be noted in the SNT model is the slightly decrease of overhead with SoCS, when compared to DiLoC. In fact, it has been shown that DiLoC is capable of clustering the same amount of nodes that SoCS. However, it uses more network resources as it presents a larger overhead. The link history kept by SoCS provides the social grouping of nodes, hence it is capable of creating more accurate clusters, resulting in overhead reduction.

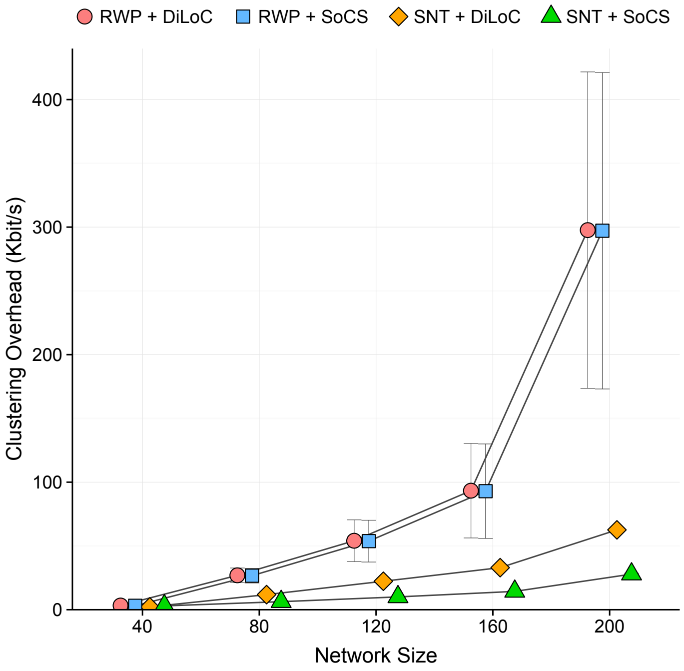


Figure 5.4: Average Clustering Overhead per second (Kbits)

5.3.2.2 Routing

The clustering structure topology strongly influences the routing performance. A good cluster structure, i.e. with a potentially low re-affiliation of nodes, provides routing a stable platform.

Thus, it is important to analyse the routing performance to assess the quality of the cluster topology.

Neighbourhood Changes or MPR Calculations A neighbourhood change occurs when a 1-hop or 2-hop node neighbour is added or deleted. Each neighbourhood change leads to a recalculation of the multipoint relay (MPR) and route table recalculation. Therefore, the number of neighbourhood changes is equal to the number of MPR calculations. Figure 5.5 depicts the average amount of neighbourhood changes per second, in the entire network. From an immediate glance, the figure shows that with the RWP model, the amount of neighbourhood changes significantly increase the with network size. Once more, this occurs due to random node movement, constantly disrupting connections. With SNT mobility, the DiLoC model has a larger amount of neighbourhood changes when compared to SoCS. Since SoCS creates and maintains clusters according to social links, the probability of having a node affiliated with a "foreign" cluster (i.e. not belonging to a given social group) is very reduced. DiLoC, lacking this information, is more prone to neighbourhood changes, specifically in this metric, as it detects 2-hop node changes.

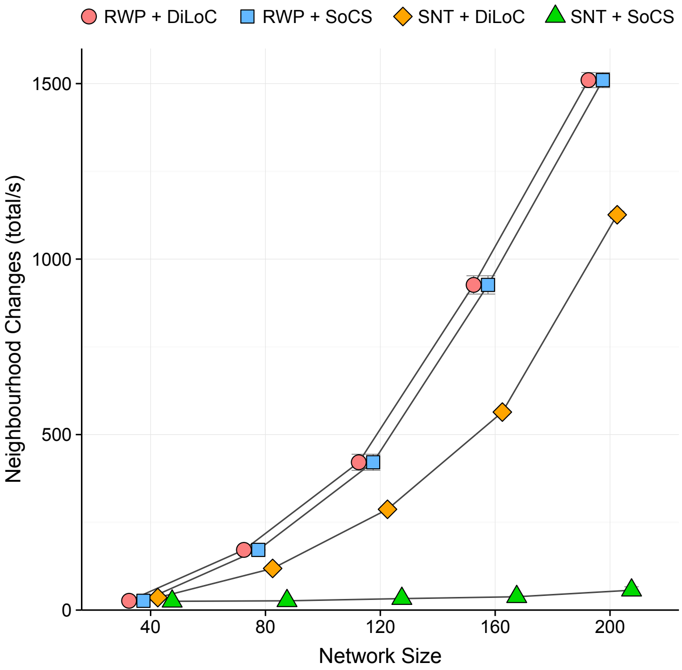


Figure 5.5: Average Amount of neighbourhood changes per second

Topology Changes In C-OLSR, upon recalculation of MPRs, topology control (TC) messages are sent and forwarded. Each received TC message leads to a topology change. The topology changes metric can be used to assess the scalability of C-OLSR within a given scenario. Figure 5.6 shows the average amount of topology changes per second, in the entire network. Once more, the RWP model shows significant higher topology changes when compared to the SNT model. Similarly to the neighbourhood changes metric, the gain related to the social link history in SoCS is very clear.

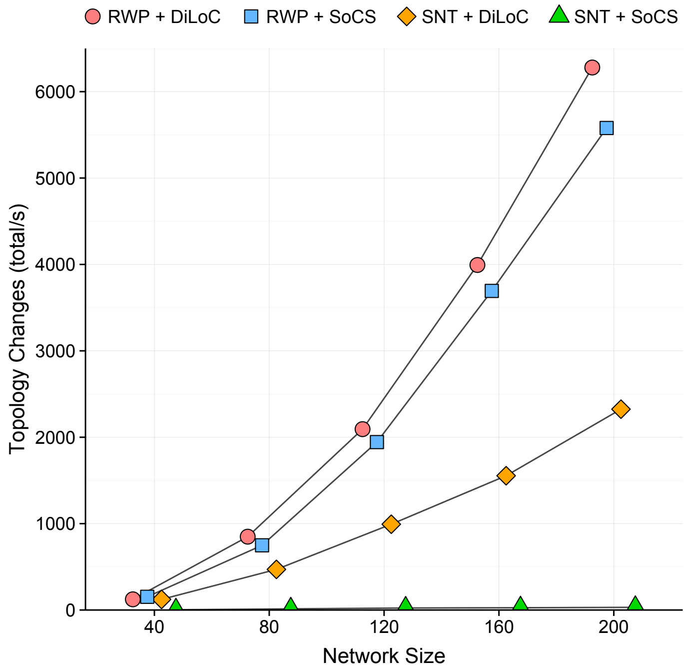


Figure 5.6: Average Amount of topology changes per second

Table 5.2: Average Amount of topology changes per second (SNT + SoCs)

	40	**80**	**120**	**160**	**200**
Mean	3.21	13.53	23.58	26.38	31.09
Std Dev	0.04	0.11	0.30	1.53	2.17

In fact, the SoCS with SNT mobility cannot be properly noticed in Figure 5.6, due to the difference on the amount of topology changes. Table 5.2 shows average amount of topology changes with SNT and SoCS.

5.3.2.3 Traffic

The analysis of traffic outlines the overall network performance. Here, it is discussed the amount of transmitted traffic, the delay and the overall network load.

Received Traffic This metric calculates the amount of successfully received traffic amongst the generated traffic. Figure 5.7 shows the average percentage of successfully received traffic per second. It is noticeable that all scenarios suffer packet loss. For networks larger than 120 nodes, the increase of packet loss is often associated with radio interferences inherent to a high density network. In smaller networks, the received traffic is also lower due to the poor connectivity associated with the low density of nodes. To be noted that in spite of most nodes in the network being clustered, it does not mean that all clusters are connected. A cluster may loose connectivity with remainder cluster, not being able to perform inter-cluster communication. Thus, since some part of the generated traffic is destined to random nodes in the network, it may not reach the destination.

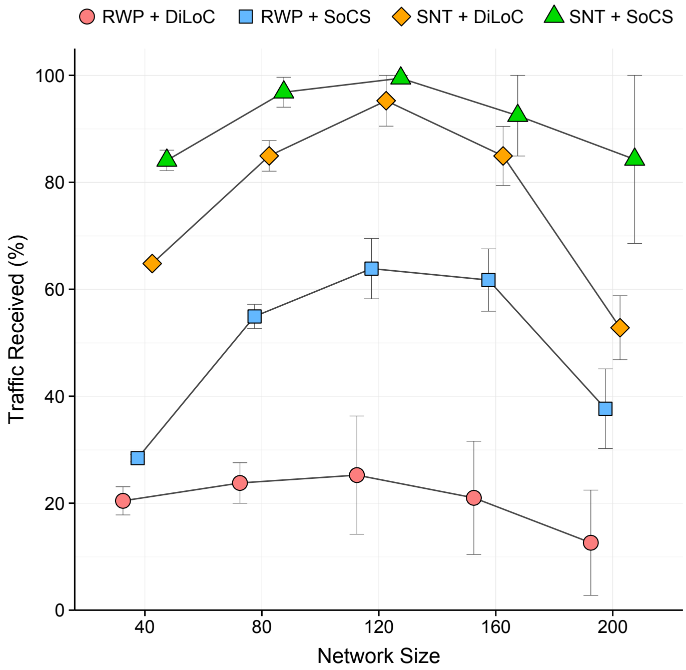


Figure 5.7: Average percentage of traffic received per second

The amount of received traffic is consistent with the obtained results in the routing performance, in both SoCS and DiLoC. The RWP model is prone to the highest amounts of packet loss and SNT the lowest. SoCS presents the highest amount of received traffic.

End2End Delay The End2End Delay measures the delay of generated packets, in seconds, in the entire network.

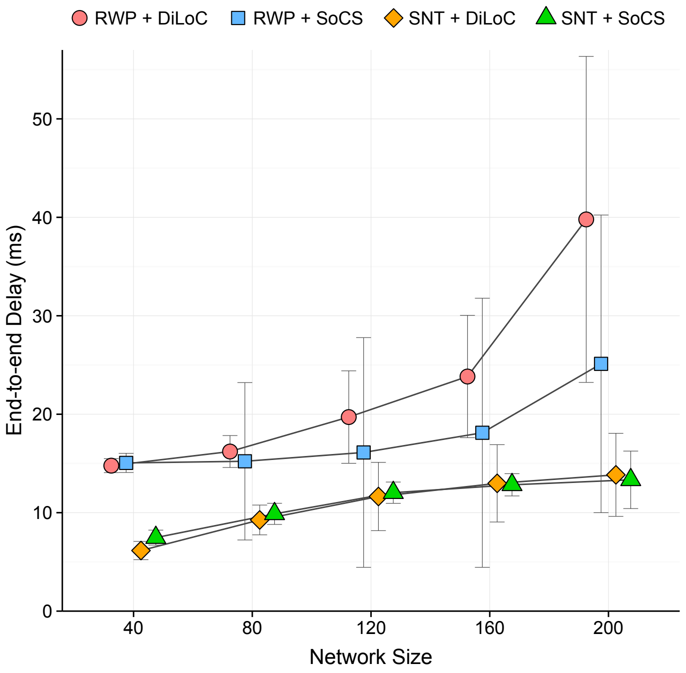


Figure 5.8: Average end-to-end delay (in milliseconds)

The elapsed time is measured between the creation of the packet at the source and its destruction at the destination. Figure 5.8 shows the average end-to-end delay for the evaluated scenarios. As depicted in the figure, the confidence intervals of delay are considerable, particularly in larger/denser networks. Once more, this is related with the radio interference inherent to wireless ad hoc networks. As expected the RWP model is prone to more delay, compared to the SNT model. There is no significant difference between the SoCS and DiLoC using the SNT model.

Network Load The network load represents the total load (in kbit/s) submitted to all nodes in the network.

This metric translates the overall weight of the network, including clustering overhead, routing overhead and generated traffic. Figure 5.9 shows the average amount of network load per second, for the entire network. Results clearly show a significant discrepancy between the RWP and SNT models. With RWP model the overhead of clustering combined with routing is around 3 times greater than with the SNT model, hence the significant difference of network load. In this metric, SoCS also shows a slight improvement when compared to

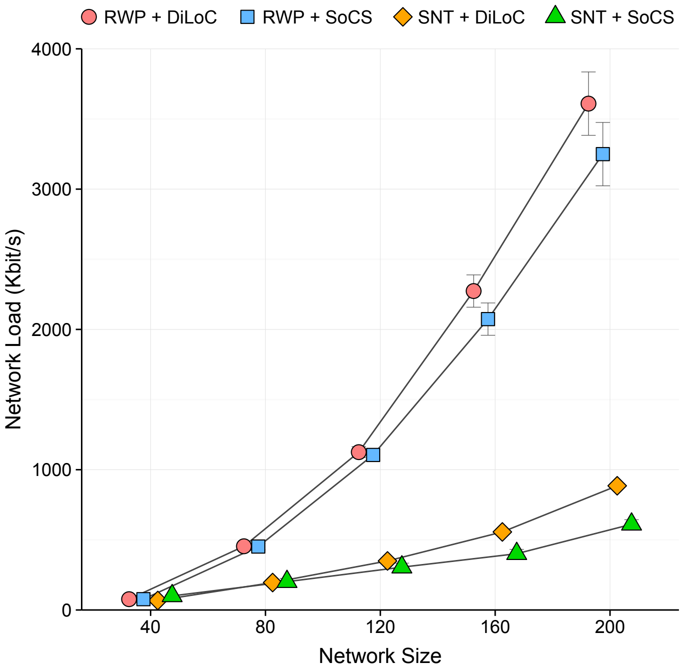


Figure 5.9: Average Network Load per second (Kbits)

DiLoC. It can also be noticed that the network is far from being saturated in all scenarios. Despite this fact, the amount of successfully received traffic (Figure 5.7) decreases for larger networks mainly due to radio interferences in denser networks, resulting in a higher packet loss.

5.4 Summary

In this chapter, the Social-aware Clustering Scheme (SoCS) clustering scheme was presented and evaluated. A list of connections with neighbours is preserved by each node, attempting to create reliable clusters, consistent with social groups. SoCS was evaluated using the C-OLSR routing protocol and compared with the DiLoC clustering scheme. To reflect the gain of social group mobility, the Random Waypoint (RWP) and the Social Theory Network (SNT) mobility models were used. Results demonstrate that SoCS provides a significant increase of network scalability, improving the routing performance and transmitted traffic. This proves that node connection history is a key feature to further improve the connection stability of wireless ad hoc networks. Social mobility particularly enhances the potential of SoCS, due to the organised dynamic of social grouping. Generally, it is shown that SoCS outperforms DiLoC in clustering, routing and traffic performance.

The development of SoCS resulted in an article entitled "Social-aware Clustering for Wireless Ad hoc Networks" published in the proceedings of the "8th Wireless Days International Conference".

Chapter 6

Mobility Modelling in Disaster Areas

Post-disaster scenarios are typically considered to exist in deserts, forests or heavily damaged urban areas, often lacking operational network infrastructures. The establishment of a temporary communication system is crucial for the assistance of victims. Mobile wireless networks are often the only capable technology to answer to this type of demands. The evaluation of the network performance for such situations in a real world scenario is, in most cases infeasible, since the cost of the repeatability of the disaster scenario would be very high and extremely difficult to reproduce. Thus, simulation evaluation allows the study of the behaviour and performance of mobile wireless networks in post-disaster environments.

The performance of mobile wireless networks strongly depend on the used mobility models [Bai et al., 2003]. Since in post-disaster environments most nodes are mobile, the used mobility model has a crucial impact on the results. Most performance evaluations existent in literature have modelled node mobility using the Random Waypoint (RWP) [Johnson and Maltz, 1996; Broch et al., 1998]. Such model is easy to implement and visualise, however it is generally unrealistic for modelling real world scenarios due to the uneven distribution of nodes. Moreover, considering the search for victim (SFV) in disaster scenarios, it becomes necessary a cooperation between mobile nodes in order to deliver an efficient area exploration. Group mobility models provide such cooperation, capable of movement coordination within groups of nodes. The Reference Point Group Mobility (RPGM) [Hong et al., 1999] combines the organised mobility of a group but also allows individual motion.

This chapter is organised as follows. Section 6.1 performs an analysis of the most relevant mobility models in literature and discusses their suitability when employed in disaster areas. Section 6.2 introduces the Human Behaviour for Disaster Areas (HBDA) mobility model, specifically designed to mimic human behaviour in disaster areas. Section 6.3 provides a performance evaluation of HBDA in comparison with the RWP and Column Mobility (COL) models. Finally, Section 6.4 concludes the chapter.

6.1 Mobility Models

Mobility models can be segmented according to node dependencies. Currently in literature there are two types of mobility models, namely *Entity Mobility Models* and *Group Mobility Models*. Entity Mobility models represent mobile nodes whose movements are independent of each other, whereas Group Mobility Models represent mobile nodes that have spatial dependencies, where the movement of a node influences the movement of at least one node around it. Regardless of their type, mobility models must meet some requirements to accurately represent movement patterns in post-disaster areas. These requirements are described as follows.

- **Coordination** - the trajectories of nodes must not be entirely random. Random mobility may exist, however some form of mobility coordination and constraints between nodes must be ensured.
- **Obstacle Avoidance** - blind exploration of the area is required, enabling nodes to avoid obstacles
- **Even Node Distribution** - it is important an even distribution of nodes to enable a full area coverage.

6.1.1 Entity Mobility Models

In this subsection several proposed entity mobility models are discussed. The Random Waypoint mobility model is the most common and used by researchers, thus its discussion is performed in more depth than the remaining.

The Random Waypoint (RWP) mobility model [Johnson and Maltz, 1996] is based on pause times between changes of direction and/or speed. Initially, nodes are placed within the scenario area in a random fashion. After deployment, nodes do not have any attachments or restrictions towards remaining nodes. Each node begins by staying in a location for a period of time. When this time expires, it travels in a random direction with a random speed [$Vmin$, $Vmax$], whereas $Vmin$ and $Vmax$ are the minimum and maximum velocity of the node, respectively. After reaching a waypoint (a decision position), the node waits another constant period of time and repeats the previous procedure until it reaches another waypoint. This process is repeated endlessly until the execution is over. Due to its simplicity, the RWP is a widely used model in research and it is the foundation for many recent mobility models.

However, it does not represent realistic movements [Kumar et al., 2010], and its use should only be considered for general purpose scenarios.

The most important problem of the RWP model is the uneven distribution of nodes since, over execution time, nodes tend to accumulate in the middle of the simulation scenario [Bettstetter et al., 2003]. Moreover, the assumption that waypoints are uniformly distributed is not feasible for most real applications. However, using different probability distributions, the RWP is able to distribute nodes impartially, highlighting certain regions of the scenario [Wang and Low, 2013]. A variation of the RWP, named Random Waypoint with Attraction Points (RWAP) [Bettstetter and Wagner, 2002] generates more realistic non-equally distributed mobility. However, the probability of a node visiting an attraction point is larger than the random choice of other points, resulting in a larger concentration of nodes in the attraction points.

The Clustered Mobility (ClusM) model [Lim et al., 2006] is very similar to the RWAP model, using RWP with attraction points to disaster areas. The main difference is that the attraction to the disaster area depends on concentration of nodes nearby. In other words, nodes have a lower probability of moving towards attraction areas where there is already a high density of nodes. Thus, in a scenario with multiple disaster areas (in this case, used as attraction points), nodes tend to be evenly distributed across those areas.

6.1.2 Group Mobility Models

The previous subsection presented the mobility models whose nodes actions are completely independent of each other. However, there are situations where nodes must mutually coordinate to achieve a certain objective, such as search and rescue operations. In order to model cooperative situations, a group mobility model is required. The Reference Point Group Mobility (RPGM) [Hong et al., 1999] can be considered a reference model, as there are many extensions and improvements of it in literature. The RPGM model allows the random motion of a group and also enables the individual motion of a node within its group. Every group has a logical centre, which controls the mobility parameters, such as motion behaviour, location, speed and direction of the entire group. Furthermore, every group is confined to a well defined geographical scope, from where its nodes can not exit. Therefore, all nodes have spatial dependencies defined by the logical centre. Sánchez et al. proposed three variations of the RPGM model in order to cover distinct objectives, namely the Column Mobility model, the Nomadic Community Mobility model and the Pursue Mobility model [Sánchez and Manzoni, 2001].

The Column Mobility (COL) can be used for search purposes. A group of mobile nodes moves in a line formation (or column) towards a random direction. Each node is tied to a reference point and each reference point is followed by another, i.e. each reference point depends on another until the head of the column is reached. Within groups, each node can move randomly around its reference point, however not exceeding a pre-configured maximum distance. The COL Model can be useful for searching purposes, whereas several groups/columns move in distinct directions and nodes move randomly inside each column. This mobility model can be obtained using a variation of the RPGM model implementation.

The Nomadic Community Mobility (NCM) is also a variation of the RPGM model. The community (or group) is defined as several nodes following only one reference point. A random direction and speed of the reference point is calculated. The group of nodes follows the reference point and can also move randomly around it, once more not exceeding a pre-configured maximum distance.

The Pursue Mobility (PM) attempts to imitate the tracking of a certain target. A group of nodes follows one particular node, adjusting their speed and direction according to the target. Within the group, nodes can move randomly but can not exceed a pre-configured distance from each other. For example, to better illustrate, this model could represent a group of police officers attempting to catch an individual. Again, this mobility model can be obtained using a modified version of the RPGM model.

The Disaster Areas (DA) model [Aschenbruck et al., 2009] mobility model is specifically designed for disaster scenarios. The model is based on the displacement of civil protection forces in real life, containing different areas according to several categories (e.g. incident site, casualties treatment area, transport zone, hospital zone). Technically, the disaster area scenario consists of several sub-areas with different configurations. Each sub-area uses a visibility graph to avoid obstacles. Each node is manually assigned to one sub-area and it is not allowed to exit unless it belongs to the transport zone sub-area. In the transport zone sub-area, nodes are allowed to leave and join, in order to represent the transportation of injured patients to the hospital. Despite the effort of mimicking a real scenario, the mobility model is still quite unrealistic as movement of rescue agents is based on the RWP mobility model, particularly in the disaster site sub-area, where agents are performing search-for-victim operations.

The Composite Mobility (CoM) model [Pomportes et al., 2010] is also designed for disaster scenarios. It is a combination of several existing models to better represent human mobility in disaster areas. The original RPGM model is used, however for better realism the RWP is replaced by the Levy-Walk model [Rhee et al., 2008]. The CoM model also concerns obstacle avoidance based on a modified Voronoï diagram. Thus, this model is based on a well known geographic map and is driven by a specific target, using the Dijkstra algorithm to calculate the shortest path between two points. However, in a disaster scenario it is very difficult to accurately obtain the current map, whereas its infrastructures may be modified or non existent. Therefore, following a known map of the area could not be sufficient to successfully perform search and rescue operations.

The Bird-Flocking Behaviour Inspired Group Mobility (BFBIGM) [Misra and Agarwal, 2011] takes inspiration from the mobility of bird flocks, flying in group coordination. In this model, several distinct groups of nodes may exist (flocks), but all have the same destination, which is a common randomly defined target. Within each group, nodes move in formation and attempt to avoid collision by keeping a safe distance between neighbours. Nodes are also capable of avoiding obstacle collision by deviating their path upon obstacle detection.

The Stream Mobility (StrM) model [Merkel et al., 2012] has a similar mobility pattern, simulating nodes in moving water or wind. Each node chooses a random angle and speed. When a node moves, it shares its angle with its neighbours. The neighbours modify the angle by adding or reducing a randomly chosen degree between 0 and 30. Thus, each node influences the movement of its neighbours.

6.1.3 Summary of Mobility Models

This section studied some of the most relevant mobility models in literature, presented in table 6.1. The Entity Mobility Models do not establish any relationship between nodes, thus not being suitable to represent movements in disaster scenarios due to the lack of node coordination. Nonetheless, the ClusM model can be considered to obtain an even distribution of nodes due to the avoidance of regions with a high density of nodes.

Table 6.1: Requirements of Mobility Models for Disaster Areas

Model	**Coordination**	**Obstacle Avoidance**	**Even Node Distribution**
RWP	✗	✗	✗
RWAP	✗	✗	✗
ClusM	✗	✗	✓
RPGM	✓	✗	✗
COL	✓	✗	✓
NCM	✓	✗	✗
PM	✓	✗	✗
DA	✓	✓	✗
CoM	✓	✓	✗
BFBIGM	✓	✓	✗
StrM	✓	✗	✗

[1] ✓: satisfies requirement ✗: does not satisfy requirement

On the other hand, all Group Mobility Models provide node coordination. The RPGM model is widely used in literature and many proposals derive from it, due to its configuration versatility. There are also a few Group Entity Models designed specifically for disaster environments. The DA mobility model is not based in random movements, since it implements specific movements between the sub-areas of the scenario. However, within each sub-area the RWP model is used, ultimately resulting in uneven node distribution within sub-areas. The CoM uses the Levy-Walk model instead. Nonetheless, a map of the post-disaster area must be known in advance, which in most situations is very difficult to obtain. The BFBIGM model does not require knowledge of the map and it is capable of blind navigation, since obstacle avoidance is supported. However, the behaviour of node mobility is not suitable for area exploration, since nodes travel from an initial point to destination as directly as possible, only deviating their path to avoid obstacle collision. In the StrM model, the moving angle of nodes is influenced by its neighbours. Ultimately, the first node to initiate movement is the leader of the group, resulting in a group of nodes following a single node.

By the analysis of Table 6.1 it can be concluded that none of the studied mobility models provide all the defined requirements suitable to realistically represent node mobility in disaster areas.

6.2 Human Behaviour for Disaster Areas

In order to obtain accurate representative results in mobile wireless networks it is necessary to use a mobility model that is capable of reproducing as much as possible a real scenario. This work is focused on post-disaster areas whereas the typical mobility pattern is based in search for victim (SFV) operations. As previously studied, most of simulation evaluations are based on the Random Waypoint model, which often does not represent the reality of node movements. The studied mobility models for disaster areas are random based, even the Disaster Areas model, which also uses the RWP within sub-areas. Aiming at overcoming the limitations of the studied models, a new mobility model was proposed driven to represent real mobility of humans in disaster environments. In this section, the Human Behaviour for Disaster Areas (HBDA) model is proposed and formally described.

6.2.1 Description

Regarding human behaviour, when a group of people is performing search operations, each person tends to physically separate from one another, in order to scout unexplored areas. On the other hand, each person typically maintains a line of sight (or in-range communicable) to at least one other person in order to be able to announce a possible victim discovery.

The group of people start the area exploration from an initial position and step-by-step, each individual makes his way to a Target Position, constantly maintaining a light of sigh to another (*maximum distance*) and, at the same time, not becoming too close (*minimum distance*). This method of search seamlessly forces individuals to evenly spread across the scenario in order to cover as much area as possible. During exploration, nodes are able avoid obstacle collision, in order to mimic real life diversion routes before reaching obstacles.

6.2.1.1 Assumptions

To accurately represent the HBDA model, a list of parameters is presented in Table 6.2.

Table 6.2: HBDA Motion Parameters

Parameter	**Description**
x and y	Generic identifier of a node position in Cartesian coordinates (x, y)
$C_i(X_i,Y_i)$	Initial coordinate position of nodes
S_x	Horizontal scenario size divided by 2. In a Cartesian system, this variable represents the horizontal length of a quadrant.
S_y	Vertical scenario size divided by 2. In a Cartesian system, this variable represents the vertical length of a quadrant.
E	Scenario Edge size
N	Number of nodes
μ	Connectivity distance or radius of nodes.
$minV, maxV$	Minimum and Maximum node velocity (in meters per second)

6.2.1.2 Start and Target Points

In the HBDA model all nodes initiate their movement from a initial coordinate $C_i(X_i, Y_i)$ located in the edges of the scenario. The determination of such coordinates is random, i.e. nodes can start at any location as long as it is located within the scenario edges, such that the condition of Equation 6.1 or Equation 6.2 is met.

$$X_i \in (|S_x - E|, |S_x|) \ \ and \ \ Y_i \in (0, |S_y|) \tag{6.1}$$

$$X_i \in (0, |S_x|) \ \ and \ \ Y_i \in (|S_y - E|, |S_y|) \tag{6.2}$$

The determination of X_i and Y_i follows a continuous uniform distribution U either on Equation 6.1 or Equation 6.2. An uniform distribution is used to deliver an equal probability to all possible coordinates. The determination of which equation to choose is based on a which follows a discrete uniform distribution on $[1, 2]$, defined in Equation 6.3.

$$\begin{gathered}|X_i'| = \begin{cases} U(S_x - E, S_x), & a = 1 \\ U(0, S_x), & a = 2 \end{cases} \\ |Y_i'| = \begin{cases} U(0, S_y), & a = 1 \\ U(S_y - E, S_y), & a = 2 \end{cases}\end{gathered} \tag{6.3}$$

Being the absolute values of X_i and Y_i determined, the Cartesian quadrant on which C_i is located is determined by q which follows a discrete uniform distribution on $[1, 4]$, represented in Equation 6.4.

$$\begin{gathered}X_i'' = \begin{cases} |X_i'|, & q = 1 \ or \ q = 4 \\ -|X_i'|, & q = 2 \ or \ q = 3 \end{cases} \\ Y_i'' = \begin{cases} |Y_i'|, & q = 1 \ or \ q = 2 \\ -|Y_i'|, & q = 3 \ or \ q = 4 \end{cases}\end{gathered} \tag{6.4}$$

The initial coordinate C_i is (X_i, Y_i), where $X_i = X_i''$ and $Y_i = Y_i''$. The destination or final coordinate of nodes $C_f(X_f, Y_f)$ is the symmetric inverse position of C_i defined in Equation 6.5.

$$C_f = (-X_i, -Y_i) \tag{6.5}$$

6.2.2 Initial Movement

The position matrix $P_{m,n}$ represents the current coordinates of all nodes. m is a constant and is equal to 2, since the mobility model is working in 2 dimensions. n is equal to the number of nodes on the scenario. Thus, every column of P represents a coordinate $(p_{x,n}, p_{y,n})$ for a node n, represented in Equation 6.6.

$$P_{2,n} = \begin{array}{c} \\ x \\ y \end{array} \begin{array}{c} \begin{array}{ccccc} Node\,1 & Node\,2 & Node\,3 & \cdots & Node\,n \end{array} \\ \begin{pmatrix} p_{x,1} & p_{x,2} & p_{x,3} & \cdots & p_{x,n} \\ p_{y,1} & p_{y,2} & p_{y,3} & \cdots & p_{y,n} \end{pmatrix} \end{array} \tag{6.6}$$

As previously mentioned, the first position of all nodes is $C_i(X_i, Y_i)$, thus $P_{x,n} = X_i$ and $P_{y,n} = Y_i, \quad \forall n \in [1, n]$.

The first movement of nodes, at time step $t = 1$, is generated randomly. For each node, a unit vector $\hat{u_i} = (u_{x_i}, u_{y_i})$ and a velocity scalar v_i are generated. Its multiplication results in a vector which will provide the first position of nodes.

6.2.2.1 Initial Unit Vectors

Each unit vector $\hat{u_i}$ is determined following a continuous uniform distribution U. As known, $||\hat{u_i}|| = 1$, thus $\sqrt{u_{i_x}^2 + u_{i_y}^2} = 1$. Decomposing, two possibilities take place, Equation 6.7 or Equation 6.8.

$$u'_{i_x} \in U(0,1) \;\; \textit{and consequently};\; u'_{i_y} = (\sqrt{1 - u_{i_x}^2}) \tag{6.7}$$

or

$$u'_{i_y} \in U(0,1) \;\; \textit{and consequently};\; u'_{i_x} = (\sqrt{1 - u_{i_y}^2}) \tag{6.8}$$

The choice between 6.7 and 6.8 follows a discrete uniform distribution p on $[1, 2]$.

Furthermore, it is necessary to determine the signal of u_{i_x} and u_{i_Y}, as defined in Equation 6.9. q follows a discrete uniform distribution on $[1, 4]$.

$$\begin{aligned} u''_{i_x} &= \begin{cases} u'_{i_x}, & q = 1 \textit{ or } q = 4 \\ -u'_{i_x}, & q = 2 \textit{ or } q = 3 \end{cases} \\ u''_{i_Y} &= \begin{cases} u'_{i_Y}, & q = 1 \textit{ or } q = 2 \\ -u'_{i_Y}, & q = 3 \textit{ or } q = 4 \end{cases} \end{aligned} \tag{6.9}$$

The result is $\hat{u_i}(u_{i_x}, u_{i_y})$ with $u_{i_x} = u''_{i_x}$ and $u_{i_y} = u''_{i_y}$.

6.2.2.2 Initial Motion Vectors

A motion vector $\overrightarrow{v_i}$ for each node i is determined in Equation 6.10.

$$\overrightarrow{v_i} = \hat{u}_i.v_i \tag{6.10}$$

whereas v_i is determined following a continuous uniform distribution on $[minV, maxV]$.

The generated vectors for all nodes are represented by the matrix $V_{m,n}$ (Equation 6.11). Similarly to the position matrix, m is a constant equal to 2 and n is the number of nodes.

$$V_{2,n} = \begin{array}{c} \\ x \\ y \end{array} \begin{array}{ccccc} Node\,1 & Node\,2 & Node\,3 & \cdots & Node\,n \\ \end{array} \begin{pmatrix} v_{x,1} & v_{x,2} & v_{x,3} & \cdots & v_{x,n} \\ v_{y,1} & v_{y,2} & v_{y,3} & \cdots & v_{y,n} \end{pmatrix} \tag{6.11}$$

Having determined the vectors the new matrix position of nodes $P_{t=1}$ is calculated by the summing the last position $P_{t=0}$ and motion vector matrices, according to Equation 6.12.

$$P_{(t=1)} = P_{(t=0)} + V \tag{6.12}$$

6.2.3 Determination of Movement

Node movement is determined at each time step t based on the current position of nodes. The coordinate of a node i at a time step t is denoted by the position matrix $P_{(t)}(p_{x,i}, p_{y,i})$.

The position of nodes for the next time step $P_{(t+1)}$ is calculated deterministically, based on two force matrices I and T and a scalar λ, according to Equation 6.13.

$$P_{(t+1)} = I_{(t)} + \lambda.T_{(t)} \tag{6.13}$$

The $I_{(t)}$ matrix represents the independence force vectors, which is calculated according to the proximity of neighbour nodes. A node n_1 is considered neighbour of node n_2 if the distance d between them is less or equal to the connectivity radius μ. If neighbour nodes do not exist, the resulting $I_{(t)}$ matrix is null. The $T_{(t)}$ matrix represents the Target (or final destination) of nodes, which is calculated based on the position of the Target. The scalar λ must be equal to one of two constants, 0 or 1, according to whether $I_{(t)}$ is not null (0) or otherwise (1). This scalar allows the exclusivity between $I_{(t)}$ and $T_{(t)}$. In other words, each node either attempts to separate from its neighbours or moves towards the Target.

6.2.3.1 Independence Force Matrix

The independence force of a node is based on the sum of the vectors towards its neighbours. In robot motion planing, obstacles represent unwanted positions in order to avoid collision [LaValle, 2011]. Thus, negative force vectors are generated towards the obstacles in order to avoid them. Similarly, the movement towards the position of neighbour nodes is undesirable, since their area is already explored.

Let the matrix $N_i(t)$ represent the position of the neighbours of each node i at each time step t.

The vectors from $P_i(t)$ to each neighbour in $N_i(t)$ are computed and represented by $\overrightarrow{Fi}(t)$.

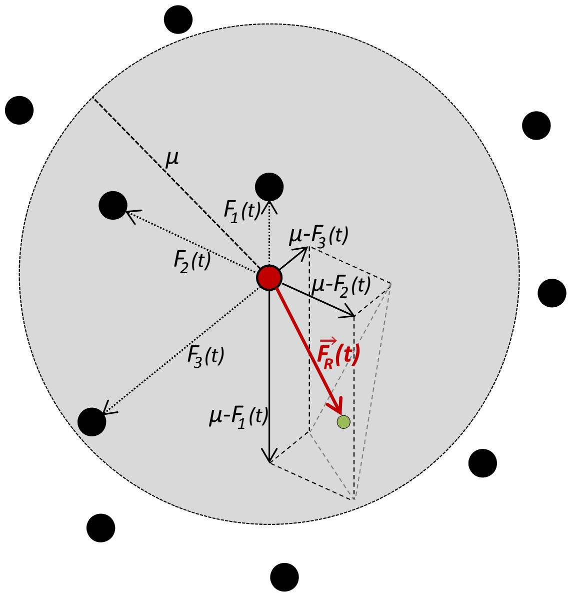


Figure 6.1: Example of a resultant force vector

At each time step t, a resultant force vector $\overrightarrow{F_R}(t)$ is computed for each node. For better illustration, Figure 6.1 shows an example of a resultant force vector $\overrightarrow{F_R}(t)$ for three neighbor nodes. Each force vector $\overrightarrow{F_R}(t)$ is calculated for the each neighbour node, according to Equation 6.14.

$$\overrightarrow{F_R}(t) = \sum_{i=1}^{m} (\mu - \overrightarrow{Fi}(t)) \tag{6.14}$$

where m is the number of neighbours, μ represents the connectivity radius and $\overrightarrow{F_i}(t)$ is the computed vector between $P_i(t)$ and each neighbour in $N_i(t)$.

The resultant vector $\overrightarrow{F_R}(t)$ cannot be applied directly to the node movement, since the velocity must be controlled between $[minV, maxV]$ and randomly generated. Thus, the unit vector $\hat{F_R}(t)$ is computed following Equation 6.15.

$$\hat{F_R}(t) = \frac{\overrightarrow{F_R}(t)}{||\overrightarrow{F_R}(t)||} \iff \hat{F_R}(t)\left(\frac{F_{R_x}}{\sqrt{F_{R_x}^2 + F_{R_y}^2}}, \frac{F_{R_y}}{\sqrt{F_{R_x}^2 + F_{R_y}^2}}\right) \tag{6.15}$$

In order to determine the motion vector, the velocity scalar v is calculated following a discrete uniform distribution on $[minV, maxV]$ and the new independence vector is determined according to Equation 6.16.

$$\overrightarrow{I_i}(t) = Pi_{(t)}.\hat{F_R}(t).v \tag{6.16}$$

The collation of all independence vectors $\overrightarrow{I_i}(t)$ results in the independence matrix $I_{(t)}$.

6.2.3.2 Target Force Matrix

In order to determine the Target force for each node, the vector towards Target t_i is calculated according to Equation 6.17.

$$\overrightarrow{t_i} = Cf - Pi_{(t)} \iff \overrightarrow{t_i}(Cf_x - P_{x,i}(t), Cf_y - P_{y,i}(t)) \tag{6.17}$$

Following this, the unit vector towards the Target is computed in Equation 6.18.

$$\hat{t_i} = \frac{\overrightarrow{t_i}}{||\overrightarrow{t_i}||} \iff \hat{t_i}\left(\frac{t_{i_x}}{\sqrt{t_{i_x}^2 + t_{i_y}^2}}, \frac{t_{i_y}}{\sqrt{t_{i_x}^2 + t_{i_y}^2}}\right) \tag{6.18}$$

Finally, the velocity scalar v is calculated following a discrete uniform distribution on $[minV, maxV]$ and the new vector towards Target is determined according to Equation 6.19.

$$\overrightarrow{T_i} = Pi_{(t)}.\hat{t_i}.v \tag{6.19}$$

The matrix $T_{(t)}$ represents the current determined forces of all nodes towards the Target.

6.2.4 Obstacle Avoidance

A node will collide with an obstacle if the vector $\overrightarrow{C}(t)$ intersects the matrix $O_{2,n}$. The vector $\overrightarrow{C}(t)$ is calculated regarding the current (t) position of the node and its future position $(t+1)$, according to Equation 6.20. Thus, if the next calculated position of a node implies an intersection with an obstacle, a collision is detected.

$$\overrightarrow{C}(t) = P_{(t+1)} - P_{(t)} \tag{6.20}$$

The matrix defined in Equation 6.21 represents a single obstacle.

$$O_{2,n} = \begin{array}{c} \\ x \\ y \end{array} \begin{array}{c} \begin{array}{ccccc} Vertex\,1 & Vertex\,2 & Vertex\,3 & \cdots & Vertex\,n \end{array} \\ \begin{pmatrix} v_{x,1} & v_{x,2} & v_{x,3} & \cdots & v_{x,n} \\ v_{y,1} & v_{y,2} & v_{y,3} & \cdots & v_{y,n} \end{pmatrix} \end{array} \tag{6.21}$$

Upon detection of obstacle collision, nodes reset their execution and assume an initial movement, as previously described in subsection 6.2.2. Similarly, an initial unit vector and an initial motion vector is generated. This behaviour enables nodes to assume a temporarily random movement, discarding the influence of the independence force and target force matrixes. In other words, upon obstacle collision detection, nodes move randomly regardless of their neighbours, in order to obtain a trajectory suitable to avoid obstacle collision. Moreover, during random movement, the collision detection mechanism remains enabled, which permanently enforces the avoidance of any collision.

6.3 Evaluation of Human Behaviour for Disaster Areas

This section presents a performance evaluation of the network, when using three distinct mobility models, namely HBDA, RWP and COL. This evaluation has the main objective of accessing how mobility models affect the performance of the network, when used in simulation evaluations.

6.3.1 Environment and Parameters

The scenario and parameter variations utilised to evaluate HBDA were selected attempting to represent, as much as possible, realistic disaster environments. In this specification the evaluation parameters were divided in four groups (Table 5.1). The General Parameters and Traffic Generation Parameters are common to the HBDA, RWP and COL models. The HBDA, RWP and COL parameters are specific to the HBDA, RWP and COL mobility models, respectively.

The conducted simulations were performed using the OPNET Modeler [OPNET Technologies Inc. (Bethesda USA), 1986]. The Free-Space propagation model was used and configured with a transmitting power and reception power threshold equivalent to a path loss distance of 150 meters. Network sizes were varied between 25 and 100 nodes in order to assess the scalability of routing for the different models. In this evaluation a proactive routing protocol was used in order to evaluate the impact of constant path establishment.

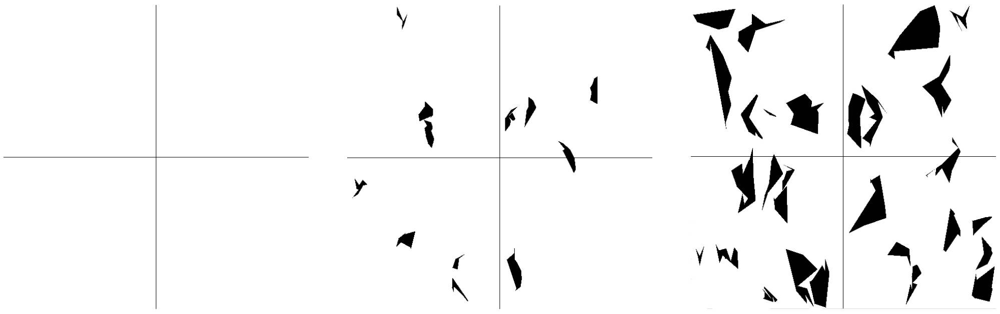
The Optimized Link State Routing protocol (OLSR) [Jacquet et al., 2001] was utilised for this purpose. To access link performance, each node generates a packet of 4096 bits every 4 seconds. Upon packet transmission its destination is a randomly chosen node in the network.

Table 6.3: General Simulation Parameters

General Parameters	
Simulator	OPNET Modeler 17.5
Simulation duration time (s)	900
RF Propagation model	Free-Space
Path Loss distance (m)	150
Network size (number of nodes)	25; 50; 75; 100
Area Size (m^2)	500 × 500
WLAN IEEE Standard	802.11g (54 Mbps)
Routing Protocol	OLSR
Mobility Model	HBDA; RWP; COL
Obstacle density	Clears; Den 1 (low); Den 2 (high)
Traffic Generation Parameters (Per Node)	
Start-Stop Time (s)	50-End of Execution
Traffic pattern	Constant Bit Rate (CBR)
Transport Protocol	User Datagram Protocol (UDP)
Packet generation rate (s)	4
Packet Size (bits)	4096
Destination Node	Random
COL Parameters	
Min-Max node speed (m/s)	1-5
Neighbour distance (m)	50
RWP Parameters	
Min-Max node speed (m/s)	1-5
Pause time (s)	None
HBDA Parameters	
Min-Max node speed (m/s)	1-5
Min-Max Distance Threshold (m)	50-100
Min-Max Travel Time	1-10

6.3.1.1 Obstacles

A post disaster scenario typically consists of an area in an unknown condition or status. Regardless of its geographic map, it becomes very challenging to obtain an accurate map of the affected area [Kleiner et al., 2007], which necessarily forces a blind mobility across the scenario. Concerning this issue, the evaluated scenarios were populated with random obstacles (different shapes and sizes) distributed randomly as well. Furthermore, in order to further assess the behaviour of mobility models, two density levels of obstacles were utilised, namely *Den 1* and *Den 2* representing a low and high density, respectively. Figure 6.2 shows examples of scenarios for the three possible obstacle densities, Clear, i.e. no obstacles, *Den 1* and *Den 2*.



(a) Clear (b) Den 1 (c) Den 2

Figure 6.2: Examples of scenarios with different obstacle densities

Since that the COL and RWP models are not capable of obstacle avoidance, it was implemented, for both, the same mechanism used in HBDA. Just before obstacle collision, the movement of node is interrupted and a new random trajectory is generated. This implemented obstacle avoidance mechanism does not significantly affect the behaviour of RWP, since it is an entity model moving according to random trajectories. However, HBDA and COL are likely to suffer from unplanned trajectories, leading to link failure between nodes.

6.3.1.2 Metrics

The metrics used to evaluate HBDA are divided in two categories, Mobility-based and Link-based. The Mobility-based metrics attempt to assess the movement characteristics produced by the mobility model. The Link-based metrics evaluate the network performance. The Mobility-based metrics are defined as follows.

- *Coverage* - represents the cumulative amount of covered area during execution time. For evaluation and comparison purposes, it has been considered that each node is able to cover 5 meters around it. It was empirically observed that a 5 meter radius is sufficiently granular in order to analyse the different coverage provided by the mobility models. Moreover, a lower value would not be realistic, since it must be considered that in SFV operations humans are able to “see” an amount of area around them. Thus, a radius of 5 meters along the trajectory of each node is considered covered.

- *Node Degree* - represents the amount of in-range nodes per node. In this evaluation, a node is considered to be in-range to another if the distance between them is at most 100 meters. Typically, in a wireless multi-hop network, a low mean node degree represents a low density network with poor connectivity [Bettstetter, 2002]. On the other hand, a high mean node degree symbolises a high density network with potential for high connectivity.

The Link-based metrics are defined as follows.

- *Topology Changes* - measures the amount of topology changes of the OLSR protocol. This metric assesses the performance of the routing protocol. Since each topology change leads to a route table recalculation, a big amount represents a poor performance efficiency.
- *Goodput* - represents the average rate at which application data packets are successfully delivered from one node to another.
- *Packet Loss* - amount of lost packets for the total amount of transmitted packets
- *Delay* - End-to-end delay of packet transmission from source to destination. The delay is measured only for the packets that arrive to the destination, i.e. successfully transmitted packets.

6.3.2 Mobility-based Results

This evaluation studies the movement characteristics of the mobility model.

6.3.2.1 Coverage

The coverage represents the amount of explored area during the movement of nodes. A higher amount of coverage lead to a higher probability of finding victims in a disaster scenario. Figure 6.3 shows the mean area coverage for the different network sizes and obstacle densities. A broad analysis of the results indicate a high amount of coverage for all models and scenarios, whereas the minimum mean of coverage is 74.10 percent. This indicates that during execution time, the trajectories of nodes provide a high coverage, even in low density scenarios with 25 nodes. Regardless, the HBDA model presents an overall higher coverage in all scenarios.

This difference is more noticeable in the 25 node network, due to the uniform dispersion of nodes provided by HBDA. On the other hand, the Column model "sweeps" the scenario with a group of nodes organised in a column fashion, which should provide a higher coverage. However, in the 25 node network, the formed column is not sufficiently wide to cover the full scenario, hence the lower coverage.

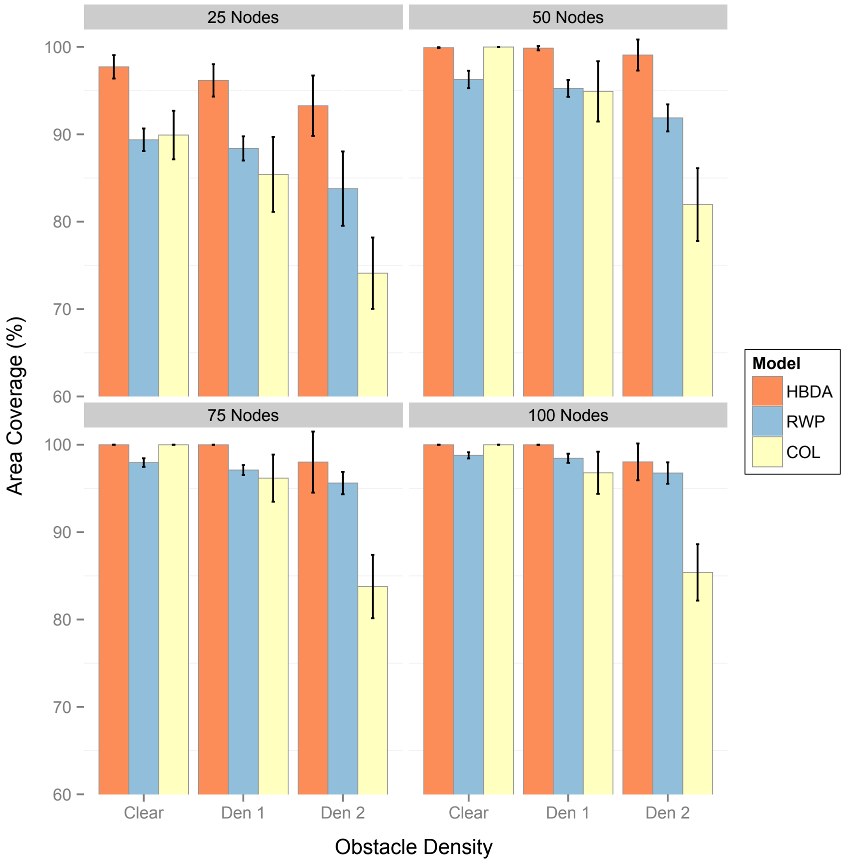


Figure 6.3: Scenario Area Coverage in percentage

The Column model offers high coverage, however it is strongly affected by obstacles, presenting significant coverage decrease in those scenarios. The Random Waypoint model is also affected by obstacles, since the premature interruption of in course random paths results in a slight lower area coverage. Finally, the HBDA is the model least affected by obstacles, presenting a consistent high coverage percentage.

6.3.2.2 Node Degree

Typically in mobile networks, a high node degree leads to a higher amount of path recalculations, which may decrease the performance of the routing protocol, consequently leading to packet loss. A high node degree may also represent good network connectivity, providing a more resilient network. However, a low node degree normally represents bad connectivity, particularly in high density scenarios. The optimal node degree for a given scenario is extremely difficult to determine, since it depends on many factors [Jerew et al., 2009], e.g. routing protocol, node density, node speed, transmission range, scenario interferences. Thus,

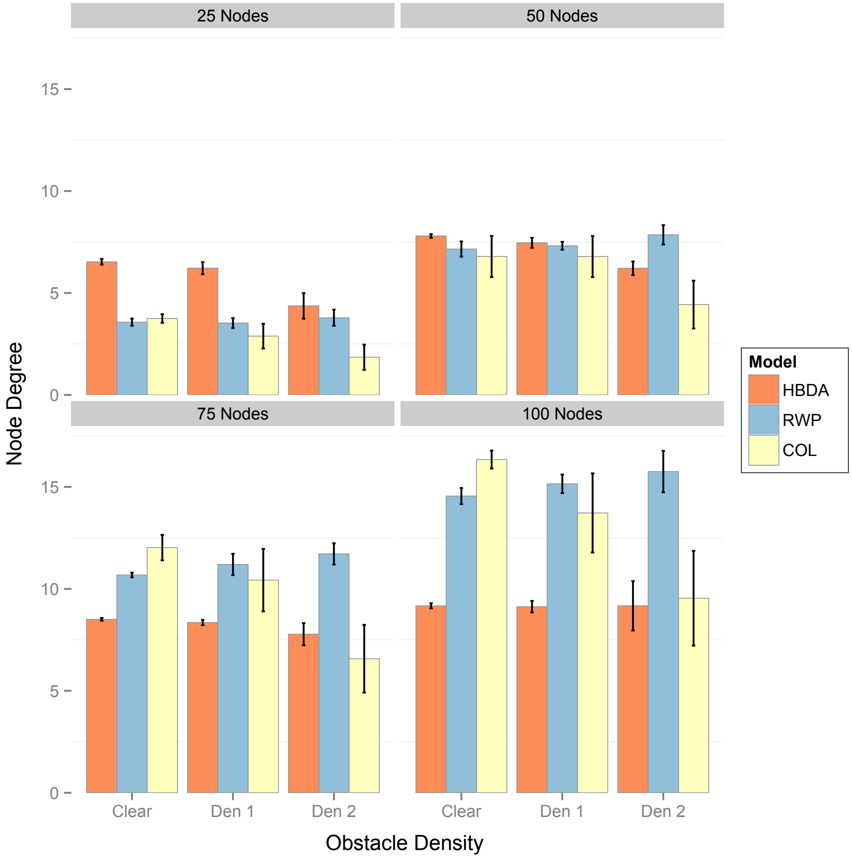


Figure 6.4: Mean node degree

the evaluation of the node degree must be based on the variation of node degree for the different scenarios. Figure 6.4 shows the node degree for the evaluated scenarios. It can be concluded that the node degree from HBDA and COL models decreases for higher obstacle densities, and the opposite occurs to RWP. In HBDA and COL models the movement of nodes is organised, i.e. nodes move according to a determined vector capable of maintaining connectivity of between nodes. Upon obstacle interference, movement vectors must be recalculated, occasionally leading to non-optimal paths, which may result in connection losses and consequently a decrease of node degree. The RWP node movement is entirely random, regardless of neighbour nodes. The increase of node degree is therefore explained due to the reduction of usable area and the increase of node density. Analysing the results across the different network sizes, there is a general increase of the node degree, particularly in the 100 nodes scenario. Moreover, the COL model is strongly affected by obstacle interference, whereas HBDA is significantly more constant in all network sizes.

6.3.3 Link-based Results

The Link-based evaluation covers the evaluation of network performance. This evaluation mainly assesses the routing efficiency for the different scenarios.

6.3.3.1 Topology Changes

In OLSR protocol each topology change leads to recalculation of the routing table, which causes network flooding. A topology change occurs when a link fails, thus highly dynamic networks typically result in a bigger amount of topology changes. From the analysis of Figure 6.5, it can be seen that RWP model causes more topology changes than the remaining

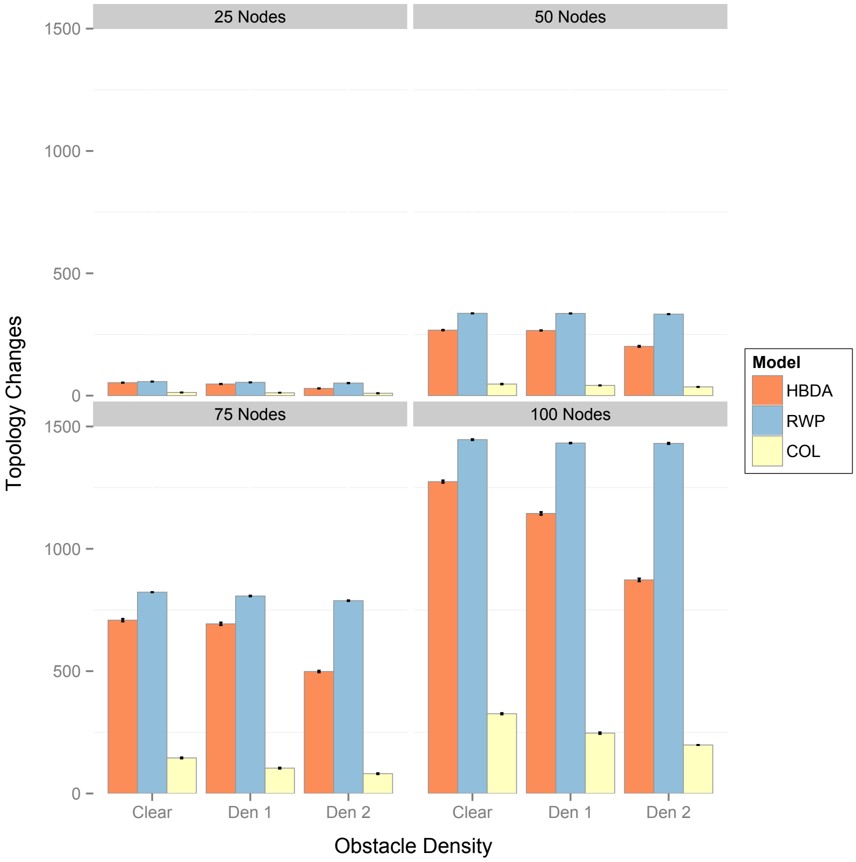


Figure 6.5: Number of Topology Changes

models. Once again, this fact is due to its characteristic uncoordinated movement, leading to constant link failures. It can also be observed in RWP model that the density of obstacles has practically no influence in the amount of topology changes. On the other hand, HBDA and COL models are significantly influenced by obstacle density.

In the presence of obstacles, HBDA and COL models tend to decrease connectivity (as evaluated in node degree) which results in smaller amounts of neighbours per node. Thus, the probability of loosing a link with a neighbour node decreases, which results in lower topology changes for higher obstacle density.
Regardless of this fact, the COL model presents the lower amount of topology changes, due to the column fashion organisation of nodes.

6.3.3.2 Goodput

The goodput represents the application level throughput, in this case the generated traffic successfully transmitted and received by each node. Figure 6.6 depicts the overall network

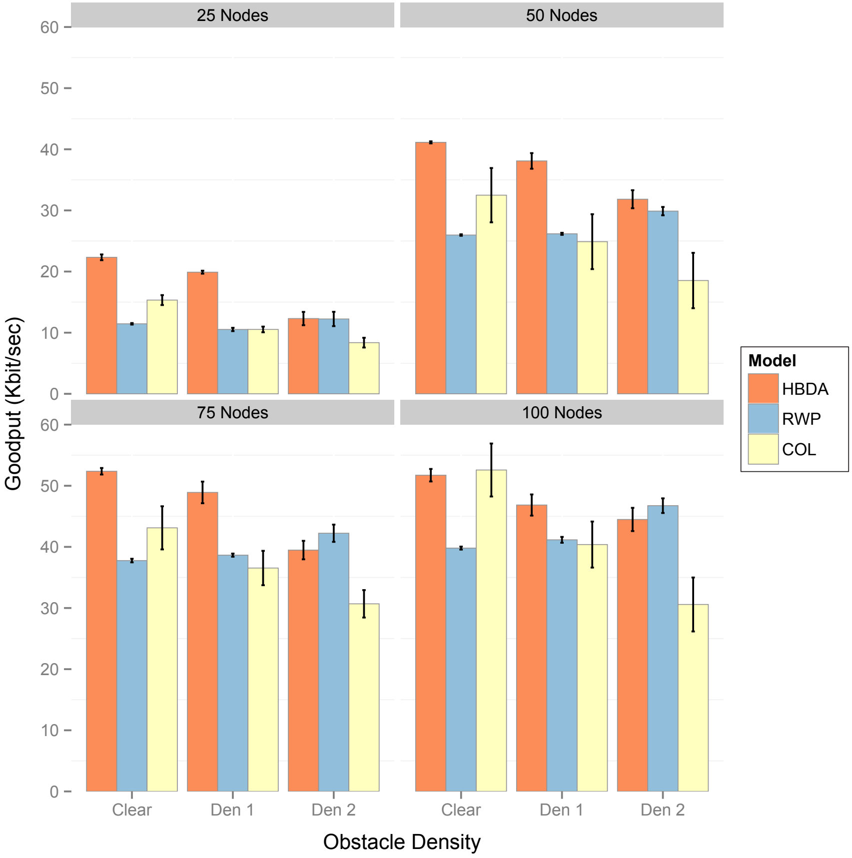


Figure 6.6: Network Goodput

goodput for the different evaluated scenarios. Similarly to the results of node degree, the RWP model presents an increase of goodput for scenarios with higher obstacle density. Once more, due to the increase of node density caused by obstacles, the RWP provides better connectivity, resulting in a higher goodput. The HBDA model provides better overall goodput. COL model is able to overcome HBDA in the 100 nodes scenario with no obstacles. Since COL model organises nodes in a column fashion, the network topology is very stable (as previously seen in the routing topology changes), reducing the routing overhead and increasing the amount of successfully transmitted data. However, the COL model heavily affected by the presence of obstacles, presenting significant lower goodput in these scenarios.

6.3.3.3 Packet Loss

The amount of lost packets is strongly tied with the goodput. Figure 6.7 shows the overall percentage of lost packets, regarding the amount of transmitted packets by all nodes. Similarly to goodput, the RWP model loses fewer packets in the presence of obstacles, whereas

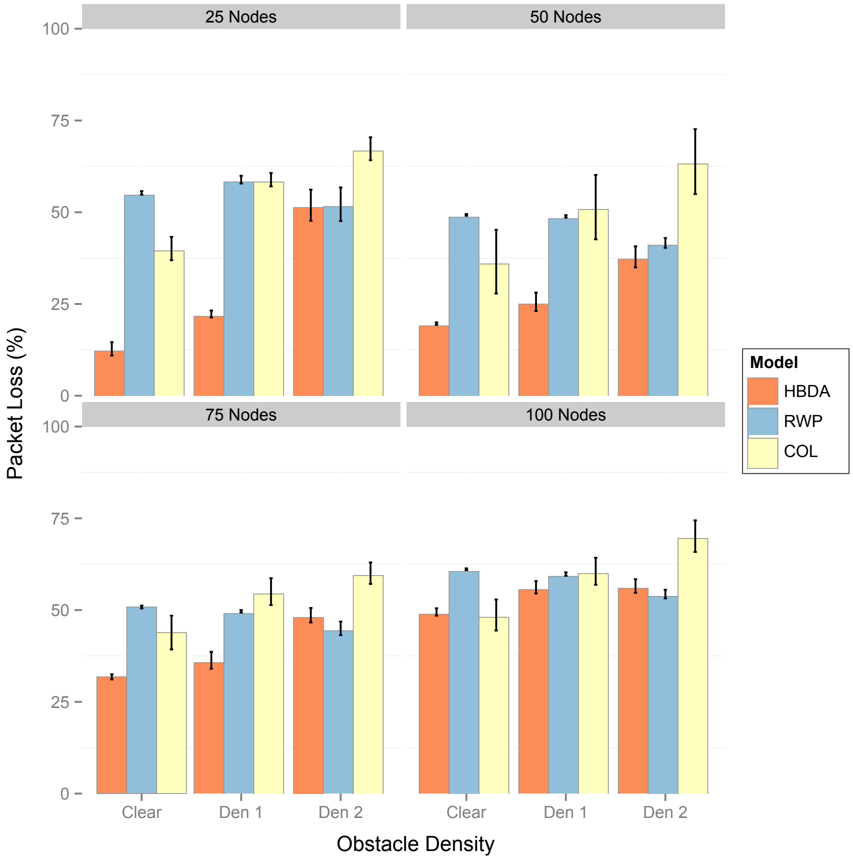


Figure 6.7: Overall Packet Loss in percentage

HBDA and COL models lose a higher amount in these scenarios. Once more, the COL model is the most affected by obstacles, reaching a 70 percent packet loss in the 100 node network with high density obstacles.

6.3.3.4 Delay

This metric measures end-to-end delay of generated application packets in seconds for the entire network between the transmission up to the reception of packets. The delay of lost packets is discarded. Packets are individually generated by each node and their destination is a random node of the network. This paradigm provides a full delay evaluation regardless of node location and distance between transmitter and receiver.

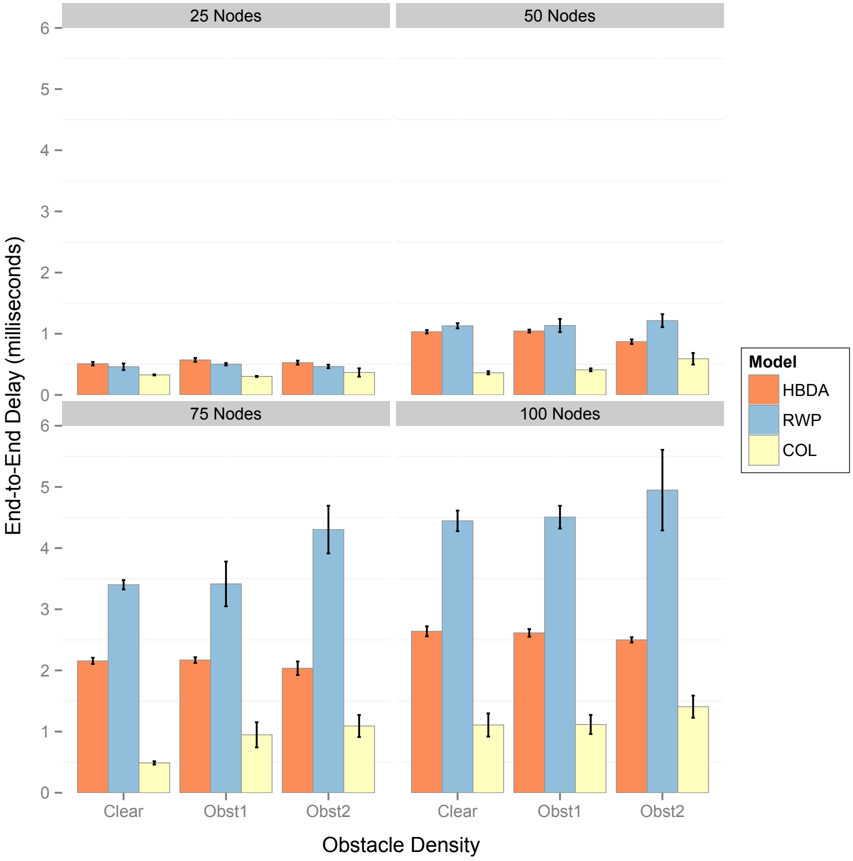


Figure 6.8: End-to-end delay of application traffic

Figure 6.8 shows the measured delay for the evaluated scenarios. Generally, the RWP model takes the biggest amount of time to deliver packets. This is caused by the constant link failure inherent to random mobility, since packets are typically stored in node queues for longer until being forwarded. Nodes were configured with a queue buffer of 256000 bits, thus it is capable of holding up to 62 packets before discarding any. Previous evaluation with lower capacity queues shown a significant increase of packet loss in RWP model. The COL model offers the less delay. Since nodes are organised in a line fashion and there is a big amount of packet loss (as previously seen) the successfully transmitted packets are often quickly delivered. HBDA sits in the average of the previous models.

6.3.4 Evaluation Summary

Table 6.4 summarises the best, average and worst results of the evaluated models in each metric. The HBDA model outperforms the remaining in most evaluation metrics, presenting a broader coverage, a more constant node degree, higher goodput and lower packet loss. The COL model, on the other hand, obtains either the best results or the worst, but never the average. Its results show less topology changes and a low end-to-end delay. As previously observed, this occurs due to the disposition of nodes in a line, keeping the network topology more stable.

Table 6.4: Summary comparison of models

Metric	**Best**	**Average**	**Worst**
Coverage	HBDA	RWP	COL
Node Degree	HBDA	RWP	COL
Topology Changes	COL	HBDA	RWP
Goodput	HBDA	RWP	COL
Packet Loss	HBDA	RWP	COL
Delay	COL	HBDA	RWP

However, in the remaining metrics, the COL model suffers to a great negative impact due to the presence of obstacles. Its coverage, node degree and goodput drop drastically in the presence of obstacles. The RWP model is never the best, obtaining in most metrics an average result.

Considering all the evaluated scenarios, HBDA is capable of wider area coverage and rate of data transmission at the cost of delay, when compared to COL. COL however, obtains very good results scenarios free from obstacles and provides fast data delivery.

6.4 Summary

In this chapter, a study of the existent mobility models was conducted, concerning necessary requirements to model post-disaster scenarios. A formal description of the HBDA model was presented covering its main mechanisms. The HBDA, RWP and COL models were evaluated, considering different types of scenarios, particularly focusing the network size and density of obstacles. In most metrics, results demonstrated significant differences between HBDA, RWP and COL. Generally, HBDA proved to be the most suitable model to handle disaster scenarios, providing better coverage efficiency and higher transmission rate. It presents, however, higher delivery delay and higher routing overhead, when compared with COL. The RWP model obtained average results in most metrics. The HBDA and COL mobility models provide a real mobility modelling for disaster scenarios, instead of random based movement decisions.

The definition of the Human Behaviour for Disaster Areas (HBDA) mobility model, resulted in a journal article entitled “Modelling Mobility based on Obstacle-aware Human Behaviour in Disaster Areas” published in Springer’s Journal “Wireless Personal Communications” and in an article entitled “Modelling Mobility based on Human Behaviour in Disaster Areas” published in the proceedings of “The 11th International Conference on Wired/Wireless Internet Communications”.

Chapter 7

Conclusion and Future Work

This chapter provides an overview of the work conducted in this thesis, including the problems that were addressed and the proposed contributions to this research topic. Section 7.1 presents a summary of the presented work and Section 7.2 discusses the open issues to be addressed in future work.

7.1 Summary of the Thesis

The clustering research topic in Mobile Ad-hoc Networks has an extensive number of contributions, with different approaches and characteristics. Chapter 2 addresses a comprehensive analysis of the most relevant clustering solutions, providing a characterisation and a discussion of the advantages and drawbacks of each scheme.

Taking into consideration the identified open issues in literature, Chapter 3 presents the Smart and Balanced Clustering scheme which provides mechanisms to tackle stability issues in large networks with high mobility. This scheme was designed to be deployed in any type of scenario, suitable for multi-purpose applications. SALSA employs two mechanisms that improve the stability of the network, the clustering balancing and an assessment of most suitable clusters to join. Clusters are balanced progressively during execution with node re-assignments, in order to maintain an even topology. Moreover, a metric to determine the most suitable cluster to join is employed which provides a preemptive mechanism to improve cluster stability.

With a more specific objective, Chapter 4 introduces the CIDNET scheme, designed to operate in indoor scenarios where WLAN infrastructures are present. The idea behind CIDNET concerns the usage of APs as location references to reduce the overhead of cluster creation while providing more uniform clusters. This solution enables the scalability of the network in areas with a high density of nodes. Moreover, Chapter 4 presents DiLoC, which overcomes the requirement for WLAN infrastructures. When no infrastructure are present, it uses special nodes, called anchor nodes, to serve as reference location points, during the initial network deployment.

Inspired by routing protocols designed for DTNs and sparse MANETs, which use social relationships information to improve forwarding decisions, the SoCS solution is proposed in Chapter 5. SoCS makes cluster assignment decisions based on social similarity of nodes. People with similar interests, location or profession are more likely to interact more often than strangers, thus the probability of being located in the same location is higher. Based on this premise, SoCS is based on connectivity frequency and connectivity durability between nodes to improve cluster management. Moreover, since social relations are dynamic, i.e. change over time, SoCS continuously updates the connection time between nodes.

One important application of Mobile Ad-hoc Networks is hazardous environments, such as post-disaster areas. The necessity of modelling these networks in such scenarios, brought the development of Human Behaviour for Disaster Areas, presented in Chapter 6. The HBDA mobility model mimics human trajectories during Search for Victim operations in post-disaster areas.

In this work, the developed clustering schemes present a different cluster management paradigm. In contrast with most schemes, clusters are created and maintained in a distributed fashion, dispensing the use of centralised nodes to regulate clusters. During the research on the proposed schemes, several advantages and drawbacks were noted. The advantages are related to the independency of using a single node to regulate a cluster. Using a centralised node often results in traffic congestion, rapid battery drainage and considerable processing load. On the other hand, distributed clustering requires a more complex management algorithm and greater overhead. As all nodes have the same role and authority, they must organise unattended to maintain the cluster structure, which typically requires an increased exchange of messages. Nonetheless, the presented paradigm was evaluated and its performance was validated for hundreds of nodes with a very reasonable overhead. The introduction of context information, such as location and social patterns, also proved to extend the clustering performance.

7.2 Future Work and Research Directions

Mobile Ad-hoc Networks are of crucial importance allowing the scalability of future networks, as discussed throughout this work. Clustering is the most feasible solution capable of enhancing the performance of routing protocols. All the proposed clustering solutions in this work are compatible to any routing protocol. In Chapter 5, the SoCS clustering scheme was evaluated with the C-OLSR, which is designed to operate in a hierarchical network. However, it would be scientifically relevant to evaluate the performance of the network with flat routing protocols, such as OLSR and AODV. In some scenarios, where long range communication is occasional, the combination of multiple routing protocols operating under the same clustering scheme may be profitable, concerning network performance. In this case, a proactive routing protocol would be responsible for intra-cluster communication and a reactive routing protocol would be in charge of inter-cluster communications. The evaluation of such scenario should be considered in a future work.

The network performance evaluation concerning the HBDA mobility model, proposed in Chapter 6, should be extended contemplating larger networks. Thus, a clustering scheme must be integrated in order to assess the network performance in post-disaster areas with a large quantity of nodes.

On a broader perspective, the number of wireless capable devices is rapidly growing worldwide. According to CISCO Visual Network Index Mobile Forecast, by 2020 there will be 11.6 billion mobile devices and connections, nearly 4 billion more than in 2015 [CISCO, 2016]. The increasingly need to support a large amount of heterogeneous devices, will result in a extensive utilisation of decentralised wireless networks. This paradigm, as of now used in the WLAN technology (wireless Ad-hoc networks), is growing in cellular communication technologies such as Long-Term Evolution Advanced (LTE-Advanced) and 5^{th} Generation (5G). Currently, several approaches to integrate Device-to-device (D2D) communications with LTE-Advanced and 5G are being researched and developed to meet the demand of future networks [Sanyal and Prasad, 2014]. Moreover, the research in D2D communications considers the utilisation of multiple technologies, such as Bluetooth, Zigbee and Wi-Fi Direct [Prasad, 2014]. This will enable decentralised networks to operate across different technologies, aiming to cover a multitude of heterogeneous devices.

References

[Abdelkader et al., 2010] Tamer Abdelkader, Kshirasagar Naik, Amiya Nayak, and Nishith Goel. A Socially-Based Routing Protocol for Delay Tolerant Networks. In *2010 IEEE Global Telecommunications Conference GLOBECOM 2010*, pages 1–5. IEEE, December 2010. ISBN 978-1-4244-5636-9. doi: 10.1109/GLOCOM.2010.5683230. URL `http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=5683230`. 88

[Amis and Prakash, 2000] A.D. Amis and R. Prakash. Load-balancing clusters in wireless ad hoc networks. In *Proceedings 3rd IEEE Symposium on Application-Specific Systems and Software Engineering Technology*, pages 25–32. IEEE Comput. Soc, 2000. ISBN 0-7695-0559-7. doi: 10.1109/ASSET.2000.888028. URL `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=888028`. 18

[Amis et al., 2000] A.D. Amis, R. Prakash, T.H.P. Vuong, and D.T. Huynh. Max-min d-cluster formation in wireless ad hoc networks. In *Proceedings IEEE INFOCOM 2000. Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (Cat. No.00CH37064)*, volume 1, pages 32–41. IEEE, 2000. ISBN 0-7803-5880-5. doi: 10.1109/INFCOM.2000.832171. URL `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=832171`. 11

[Aschenbruck et al., 2009] Nils Aschenbruck, Elmar Gerhards-Padilla, and Peter Martini. Modeling mobility in disaster area scenarios. *Performance Evaluation*, 66(12):773 – 790, 2009. ISSN 0166-5316. doi: http://dx.doi.org/10.1016/j.peva.2009.07.009. Performance Evaluation of Wireless Ad Hoc, Sensor and Ubiquitous Networks. 108

[Bahl and Padmanabhan, 2000] P. Bahl and V.N. Padmanabhan. Radar: an in-building rf-based user location and tracking system. In *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceed-*

ings. IEEE, volume 2, pages 775 –784 vol.2, 2000. doi: 10.1109/INFCOM.2000.832252. 59, 60

[Bai et al., 2003] Fan Bai, N. Sadagopan, and A. Helmy. Important: a framework to systematically analyze the impact of mobility on performance of routing protocols for adhoc networks. In *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, volume 2, pages 825–835 vol.2, 2003. doi: 10.1109/INFCOM.2003.1208920. 105

[Basu et al., 2001] P. Basu, N. Khan, and T.D.C. Little. A mobility based metric for clustering in mobile ad hoc networks. In *Proceedings 21st International Conference on Distributed Computing Systems Workshops*, pages 413–418. IEEE Comput. Soc, April 2001. ISBN 0-7695-1080-9. doi: 10.1109/CDCS.2001.918738. URL `http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=918738`. 15

[Bettstetter et al., 2003] C. Bettstetter, G. Resta, and P. Santi. The node distribution of the random waypoint mobility model for wireless ad hoc networks. *Mobile Computing, IEEE Transactions on*, 2(3):257–269, July 2003. ISSN 1536-1233. doi: 10.1109/TMC.2003.1233531. 107

[Bettstetter, 2002] Christian Bettstetter. On the minimum node degree and connectivity of a wireless multihop network. In *Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing - MobiHoc '02*, page 80, New York, New York, USA, June 2002. ACM Press. ISBN 1581135017. doi: 10.1145/513800.513811. 120

[Bettstetter and Wagner, 2002] Christian Bettstetter and Christian Wagner. The spatial node distribution of the random waypoint mobility model. In *Mobile Ad-Hoc Netzwerke, 1. Deutscher Workshop ÙBer Mobile Ad-Hoc Netzwerke WMAN 2002*, pages 41–58. GI, 2002. ISBN 3-88579-341-5. 107

[Bigwood and Henderson, 2008] Greg Bigwood and Tristan Henderson. Social dtn routing. In *Proceedings of the 2008 ACM CoNEXT Conference*, CoNEXT '08, pages 35:1–35:2, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-210-8. doi: 10.1145/1544012.1544047. URL `http://doi.acm.org/10.1145/1544012.1544047`. 89

[Bras et al., 2010] L. Bras, M. Oliveira, N. Borges de Carvalho, and P. Pinho. Low power location protocol based on zigbee wireless sensor networks. In *Indoor Positioning and Indoor Navigation (IPIN), 2010 International Conference on*, pages 1 –7, sept. 2010. doi: 10.1109/IPIN.2010.5647869. 61

[Broch et al., 1998] Josh Broch, David A. Maltz, David B. Johnson, Yih-Chun Hu, and Jorjeta Jetcheva. A performance comparison of multi-hop wireless ad hoc network routing protocols. In *Proceedings of the 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking*, MobiCom '98, pages 85–97, New York, NY, USA, 1998. ACM. ISBN 1-58113-035-X. doi: 10.1145/288235.288256. 105

[Chatterjee et al., 2000] M. Chatterjee, S.K. Sas, and D. Turgut. An on-demand weighted clustering algorithm (WCA) for ad hoc networks. In *Globecom '00 - IEEE. Global Telecommunications Conference. Conference Record (Cat. No.00CH37137)*, volume 3, pages 1697–1701. IEEE, 2000. ISBN 0-7803-6451-1. doi: 10.1109/GLOCOM.2000.891926. URL `http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=891926`. 20

[Cheng et al., 2009] Chu-Sung Cheng, H.H. Chang, Yu-Ting Chen, T.H. Lin, P.C. Chen, C.M. Huang, H.S. Yuan, and W.C. Chu. Accurate location tracking based on active rfid for health and safety monitoring. In *Bioinformatics and Biomedical Engineering , 2009. ICBBE 2009. 3rd International Conference on*, pages 1 –4, june 2009. doi: 10.1109/ICBBE.2009.5162266. 61

[Cheng et al., 2013] X. Cheng, X. Huang, and D.Z. Du. *Ad Hoc Wireless Networking*. Network Theory and Applications. Springer US, 2013. ISBN 9781461302230. URL `https://books.google.pt/books?id=q5HbBwAAQBAJ`. 30

[Cheng, 2009] Yuh-Ming Cheng. Using zigbee and room-based location technology to constructing an indoor location-based service platform. In *Intelligent Information Hiding and Multimedia Signal Processing, 2009. IIH-MSP '09. Fifth International Conference on*, pages 803 –806, sept. 2009. doi: 10.1109/IIH-MSP.2009.106. 61

[Chiang et al., 1998] Ching-chuan Chiang, Hsiao-kuang Wu, Winston Liu, and Mario Gerla. Routing In Clustered Multihop, Mobile Wireless Networks With Fading Channel. September 1998. URL `http://www.researchgate.net/publication/2284528_Routing_In_Clustered_Multihop_Mobile_Wireless_Networks_With_Fading_Channel`. 11, 15

[Ching-Chuan et al., 1997] Chiang Ching-Chuan, Wu Hsiao-Kuang, Liu Winston, and Gerla Mario. Routing in clustered multihop mobile wireless networks with fading channel. In *Proceedings of IEEE SICON*, pages 197–211, April 1997. 30

[Choi and Woo, 2006] Wonchang Choi and Miae Woo. A distributed weighted clustering algorithm for mobile ad hoc networks. *Telecommunications, 2006. AICT-*

ICIW '06. International Conference on Internet and Web Applications and Services/Advanced International Conference on, pages 73–73, Feb. 2006. doi: 10.1109/AICT-ICIW.2006.11. 23

[CISCO, 2016] CISCO. Cisco vni forecast 2016. `http://www.cisco.com/c/en/us/solutions/service-provider/visual-networking-index-vni/`, 2016. 131

[Clausen and P. Jacquet, 2003] Ed. Clausen and Ed. P. Jacquet. Optimized link state routing protocol (olsr). *IETF Request For Comments 3626*, 2003. URL `http://www.ietf.org/rfc/rfc3626.txt`. 26

[Conceição et al., 2010] Luís Conceição, David Palma, and Marilia Curado. A novel stable and low-maintenance clustering scheme. In *Proceedings of the 2010 ACM Symposium on Applied Computing*, SAC '10, pages 699–705, New York, NY, USA, 2010. ACM. ISBN 978-1-60558-639-7. doi: http://doi.acm.org/10.1145/1774088.1774232. 47

[Cruz et al., 2011] O. Cruz, E. Ramos, and M. Ramirez. 3d indoor location and navigation system based on bluetooth. In *Electrical Communications and Computers (CONIELECOMP), 2011 21st International Conference on*, pages 271 –277, 28 2011-march 2 2011. doi: 10.1109/CONIELECOMP.2011.5749373. 60

[Cunha et al., 2014] Felipe D. Cunha, Guilherme G. Maia, Aline C. Viana, Raquel A. Mini, Leandro A. Villas, and Antonio A. Loureiro. Socially inspired data dissemination for vehicular ad hoc networks. In *Proceedings of the 17th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems - MSWiM '14*, pages 81–85, New York, New York, USA, September 2014. ACM Press. ISBN 9781450330305. doi: 10.1145/2641798.2641834. URL `http://dl.acm.org/citation.cfm?id=2641798.2641834`. 88

[Daly and Haahr, 2007] Elizabeth M. Daly and Mads Haahr. Social network analysis for routing in disconnected delay-tolerant MANETs. In *Proceedings of the 8th ACM international symposium on Mobile ad hoc networking and computing - MobiHoc '07*, page 32, New York, New York, USA, September 2007. ACM Press. ISBN 9781595936844. doi: 10.1145/1288107.1288113. URL `http://dl.acm.org/citation.cfm?id=1288107.1288113`. 88, 89

[Divecha et al., 2007] Bhavyesh Divecha, Ajith Abraham, Crina Grosan, and Sugata Sanyal. Impact of node mobility on manet routing protocols models, 2007. 46

[Eagle et al., 2009] Nathan Eagle, Alex (Sandy) Pentland, and David Lazer. Inferring social network structure using mobile phone data. In *PROCEEDINGS OF NATIONAL ACADEMY OF SCIENCES*, page 2009, 2009. 87

[Ephremides et al., 1987] A. Ephremides, J.E. Wieselthier, and D.J. Baker. A design concept for reliable mobile radio networks with frequency hopping signaling. *Proceedings of the IEEE*, 75(1):56–73, January 1987. ISSN 0018-9219. doi: 10.1109/PROC.1987.13705. URL `http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=1457973`. 11

[Er and Seah, 2004] I.I. Er and W.K.G. Seah. Mobility-based d-hop clustering algorithm for mobile ad hoc networks. In *2004 IEEE Wireless Communications and Networking Conference (IEEE Cat. No.04TH8733)*, volume 4, pages 2359–2364. IEEE, 2004. ISBN 0-7803-8344-3. doi: 10.1109/WCNC.2004.1311457. URL `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1311457`. 15

[Fowler et al., 2014] S. Fowler, Yuan Li, A. Pollastro, and S. Napoli. Simple network design and power allocation for 5g device-to-device communication. In *Computer Aided Modeling and Design of Communication Links and Networks (CAMAD), 2014 IEEE 19th International Workshop on*, pages 203–207, Dec 2014. doi: 10.1109/CAMAD.2014.7033235. 1

[Freeman, 1978] Linton C. Freeman. Centrality in social networks conceptual clarification. *Social Networks*, 1(3):215–239, January 1978. ISSN 03788733. doi: 10.1016/0378-8733(78)90021-7. URL `http://www.sciencedirect.com/science/article/pii/0378873378900217`. 87

[Gerla, 2002] M. Gerla. Scalable routing protocols for mobile ad hoc networks. *IEEE Network*, 16(4):11–21, July 2002. ISSN 0890-8044. doi: 10.1109/MNET.2002.1020231. URL `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1020231`. 8

[Gerla and Tzu-Chieh Tsai, 1995] Mario Gerla and Jack Tzu-Chieh Tsai. Multicluster, mobile, multimedia radio network. *Wireless Networks*, 1(3):255–265, September 1995. ISSN 1022-0038. doi: 10.1007/BF01200845. URL `http://link.springer.com/10.1007/BF01200845`. 12

[Guillemette et al., 2008] M.G. Guillemette, I. Fontaine, and C. Caron. Hybrid rfid-gps real-time location system for human resources: Development, impacts and perspectives. In *Hawaii International Conference on System Sciences, Proceedings of the 41st Annual*, page 406, jan. 2008. doi: 10.1109/HICSS.2008.195. 62

[Gupta and Kumar, 2000] P. Gupta and P.R. Kumar. The capacity of wireless networks. *IEEE Transactions on Information Theory*, 46(2):388–404, March 2000. ISSN 00189448. doi: 10.1109/18.825799. URL `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=825799`. 8

[Haas et al., 2002] Zygmunt J. Haas, Marc R. Pearlman, and Prince Samar. The zone routing protocol (zrp) for ad hoc networksmarc r. pearlman. *IETF MANET Working Group Internet Draft (draft-ietf-manet-zone-zrp-04)*, 2002. 29

[Harter et al., 2002] Andy Harter, Andy Hopper, Pete Steggles, Andy Ward, and Paul Webster. The anatomy of a context-aware application. *Wirel. Netw.*, 8:187–197, March 2002. ISSN 1022-0038. doi: http://dx.doi.org/10.1023/A:1013767926256. 62

[Hassan et al., 2010] N.S.A. Hassan, S. Hossain, N.H.A. Wahab, S.H.S. Ariffin, N. Fisal, L.A. Latiff, M. Abbas, and Choong Khong Neng. An indoor 3d location tracking system using rssi. In *Signal-Image Technology and Internet-Based Systems (SITIS), 2010 Sixth International Conference on*, pages 323 –328, dec. 2010. doi: 10.1109/SITIS.2010.59. 59

[Hong et al., 1999] Xiaoyan Hong, Mario Gerla, Guangyu Pei, and Ching-Chuan Chiang. A group mobility model for ad hoc wireless networks. In *Proceedings of the 2Nd ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, MSWiM '99, pages 53–60, New York, NY, USA, 1999. ACM. ISBN 1-58113-173-9. doi: 10.1145/313237.313248. 105, 107

[Hsu and Feng, 2007] Chung-Hsien Hsu and Kai-Ten Feng. On-demand routing-based clustering protocol for mobile ad hoc networks. *Personal, Indoor and Mobile Radio Communications, 2007. PIMRC 2007. IEEE 18th International Symposium on*, pages 1–5, Sept. 2007. doi: 10.1109/PIMRC.2007.4394482. 34

[Huang et al., 2006] Chuanhe Huang, Yuanyuan Zhang, Xiaohua Jia, Wenming Shi, Yong Cheng, and Hao Zhou. An on-demand clustering mechanism for hierarchical routing protocol in ad hoc networks. *Wireless Communications, Networking and Mobile Computing, 2006. WiCOM 2006.International Conference on*, pages 1–6, Sept. 2006. doi: 10.1109/WiCOM.2006.217. 34

[Hui et al., 2011] Pan Hui, Jon Crowcroft, and Eiko Yoneki. BUBBLE Rap: Social-Based Forwarding in Delay-Tolerant Networks. *IEEE Transactions on Mobile Computing*, 10(11):1576–1589, November 2011. ISSN 1536-1233. doi:

10.1109/TMC.2010.246. URL `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5677535`. 85, 90

[Hussein et al., 2010] A. Hussein, S. Yousef, S. Al-Khayatt, and O.S. Arabeyyat. An efficient weighted distributed clustering algorithm for mobile ad hoc networks. In *Computer Engineering and Systems (ICCES), 2010 International Conference on*, pages 221 –228, 30 2010-dec. 2 2010. doi: 10.1109/ICCES.2010.5674857. 24

[I Selvam, 2011] V. Palanisamy I Selvam. Stable and Flexible Weight based Clustering Algorithm in Mobile Ad hoc Networks. 2011. URL `http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.206.2120`. 13

[Jacquet et al., 2001] P. Jacquet, P. Muhlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Viennot. Optimized link state routing protocol for ad hoc networks. In *Multi Topic Conference, 2001. IEEE INMIC 2001. Technology for the 21st Century. Proceedings. IEEE International*, pages 62–68, 2001. doi: 10.1109/INMIC.2001.995315. 26, 118

[Jahanbakhsh and Hajhosseini, 2008] Kazem Jahanbakhsh and Marzieh Hajhosseini. Improving Performance of Cluster Based Routing Protocol using Cross-Layer Design. February 2008. URL `http://arxiv.org/abs/0802.0543`. 33

[Jerew et al., 2009] Oday D. Jerew, Haley M. Jones, and Kim L. Blackmore. On the minimum number of neighbours for good routing performance in MANETs. In *2009 IEEE 6th International Conference on Mobile Adhoc and Sensor Systems*, pages 573–582. IEEE, October 2009. ISBN 978-1-4244-5114-2. doi: 10.1109/MOBHOC.2009.5336952. 122

[Jiangsheng, 2002] Yu Jiangsheng. Method of k-nearest neighbors. 2002. 60

[Joa-Ng and Lu, 1999] M. Joa-Ng and I-Tai Lu. A peer-to-peer zone-based two-level link state routing for mobile ad hoc networks. *Selected Areas in Communications, IEEE Journal on*, 17:1415–1425, 1999. ISSN 0733-8716. 29

[Johnson et al., 2007] D. Johnson, Rice University, Y. Hu, UIUC, D. Maltz, and Microsoft Research. The dynamic source routing protocol (dsr) for mobile ad hoc networks for ipv4. *IETF Request For Comments 4728*, 2007. URL `http://www.ietf.org/rfc/rfc4728.txt`. 28

[Johnson and Maltz, 1996] DavidB. Johnson and DavidA. Maltz. Dynamic source routing in ad hoc wireless networks. In Tomasz Imielinski and HenryF. Korth, editors, *Mobile Computing*, volume 353 of *The Kluwer International Series in Engineering and Computer Science*, pages 153–181. Springer US, 1996. ISBN 978-0-7923-9697-0. doi: 10.1007/978-0-585-29603-6_5. 105, 106

[Kahng and Goto, 2004] Hyun-Kook Kahng and Shigeki Goto, editors. *Information Networking. Networking Technologies for Broadband and Mobile Networks*, volume 3090 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004. ISBN 978-3-540-23034-2. doi: 10.1007/b99472. URL `http://www.springerlink.com/index/10.1007/b99472`. 22

[Kleiner et al., 2007] Alexander Kleiner, Christian Dornhege, and Sun Dali. Mapping disaster areas jointly: RFID-Coordinated SLAM by Hurnans and Robots. In *2007 IEEE International Workshop on Safety, Security and Rescue Robotics*, pages 1–6. IEEE, September 2007. ISBN 978-1-4244-1568-7. doi: 10.1109/SSRR.2007.4381263. 119

[Konstantopoulos et al., 2008] Charalampos Konstantopoulos, Damianos Gavalas, and Grammati Pantziou. Clustering in mobile ad hoc networks through neighborhood stability-based mobility prediction. *Computer Networks*, 52(9):1797–1824, June 2008. ISSN 13891286. doi: 10.1016/j.comnet.2008.01.018. URL `http://www.sciencedirect.com/science/article/pii/S1389128608000923`. 16

[Kumar et al., 2010] Santosh Kumar, S C Sharma, and Bhupendra Suman. Mobility Metrics Based Classification & Analysis of Mobility Model for Tactical Network. *International Journal of Next-Generation Networks*, 2(3):39–51, September 2010. ISSN 09757252. doi: 10.5121/ijngn.2010.2305. 107

[Lau and Chung, 2007] Erin-Ee-Lin Lau and Wan-Young Chung. Enhanced rssi-based real-time user location tracking system for indoor and outdoor environments. In *Proceedings of the 2007 International Conference on Convergence Information Technology*, ICCIT '07, pages 1213–1218, Washington, DC, USA, 2007. IEEE Computer Society. ISBN 0-7695-3038-9. doi: http://dx.doi.org/10.1109/ICCIT.2007.199. 60

[LaValle, 2011] S.M. LaValle. Motion planning. *Robotics Automation Magazine, IEEE*, 18 (1):79–89, March 2011. ISSN 1070-9932. doi: 10.1109/MRA.2011.940276. 115

[Lei et al., 2012] Lei Lei, Zhangdui Zhong, Chuang Lin, and Xuemin Shen. Operator controlled device-to-device communications in lte-advanced networks. *Wireless Communications, IEEE*, 19(3):96–104, June 2012. ISSN 1536-1284. doi: 10.1109/MWC.2012.6231164. 1

[Li et al., 2010] Qinghua Li, Sencun Zhu, and Guohong Cao. Routing in Socially Selfish Delay Tolerant Networks. In *2010 Proceedings IEEE INFOCOM*, pages 1–9. IEEE, March 2010. ISBN 978-1-4244-5836-3. doi: 10.1109/INFCOM.2010.5462138. URL `http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=5462138`. 85, 88

[Lim et al., 2006] Sunho Lim, Chansu Yu, and Chita Das. Clustered Mobility Model for Scale-Free Wireless Networks. *Proceedings. 2006 31st IEEE Conference on Local Computer Networks*, pages 231–238, November 2006. ISSN 0742-1303. doi: 10.1109/LCN.2006.322105. 107

[Lindgren et al., 2003] Anders Lindgren, Avri Doria, and Olov Schelén. Probabilistic routing in intermittently connected networks. *SIGMOBILE Mob. Comput. Commun. Rev.*, 7(3):19–20, July 2003. ISSN 1559-1662. doi: 10.1145/961268.961272. URL `http://doi.acm.org/10.1145/961268.961272`. 90

[Mai and Choo, 2008] Khac Tiep Mai and Hyunseung Choo. Connectivity-based clustering scheme for mobile ad hoc networks. *Research, Innovation and Vision for the Future, 2008. RIVF 2008. IEEE International Conference on*, pages 191–197, July 2008. doi: 10.1109/RIVF.2008.4586354. 13

[Mei and Stefa, 2012] Alessandro Mei and Julinda Stefa. Give2get: Forwarding in social mobile wireless networks of selfish individuals. *IEEE Transactions on Dependable and Secure Computing*, 9(4):568–581, 2012. ISSN 1545-5971. doi: http://doi.ieeecomputersociety.org/10.1109/TDSC.2012.37. 89

[Mei et al., 2011] Alessandro Mei, Giacomo Morabito, Paolo Santi, and Julinda Stefa. Social-aware stateless forwarding in pocket switched networks. In *2011 Proceedings IEEE INFOCOM*, pages 251–255. IEEE, April 2011. ISBN 978-1-4244-9919-9. doi: 10.1109/INFCOM.2011.5935076. URL `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5935076`. 85, 89

[Merkel et al., 2012] Sabrina Merkel, Sanaz Mostaghim, and Hartmut Schmeck. A Study of Mobility in Ad Hoc Networks and Its Effects on a Hop Count Based Distance Estimation. In *2012 5th International Conference on New Technologies, Mobility and Security (NTMS)*, pages 1–5. IEEE, May 2012. ISBN 978-1-4673-0229-6. doi: 10.1109/NTMS.2012.6208681. 109

[Misra and Agarwal, 2011] Sudip Misra and Prateek Agarwal. Bio-inspired group mobility model for mobile ad hoc networks based on bird-flocking behavior. *Soft Comput-*

ing, 16(3):437–450, May 2011. ISSN 1432-7643. doi: 10.1007/s00500-011-0728-x. 109

[Mtibaa et al., 2010] Abderrahmen Mtibaa, Martin May, Christophe Diot, and Mostafa Ammar. PeopleRank: Social Opportunistic Forwarding. In *2010 Proceedings IEEE INFOCOM*, pages 1–5. IEEE, March 2010. ISBN 978-1-4244-5836-3. doi: 10.1109/INFCOM.2010.5462261. URL `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5462261`. 88

[Musolesi and Mascolo, 2007] Mirco Musolesi and Cecilia Mascolo. Designing mobility models based on social network theory. *ACM SIGMOBILE Mobile Computing and Communications Review*, 11(3):59, July 2007. ISSN 15591662. doi: 10.1145/1317425.1317433. URL `http://dl.acm.org/citation.cfm?id=1317425.1317433`. 90

[Musolesi and Mascolo, 2009] Mirco Musolesi and Cecilia Mascolo. Mobility models for systems evaluation. In Benot Garbinato, Hugo Miranda, and Lus Rodrigues, editors, *Middleware for Network Eccentric and Mobile Applications*, pages 43–62. Springer Berlin Heidelberg, 2009. ISBN 978-3-540-89706-4. doi: 10.1007/978-3-540-89707-1_3. URL `http://dx.doi.org/10.1007/978-3-540-89707-1_3`. 90

[Ni et al., 2003] L.M. Ni, Yunhao Liu, Yiu Cho Lau, and A.P. Patil. Landmarc: indoor location sensing using active rfid. In *Pervasive Computing and Communications, 2003. (PerCom 2003). Proceedings of the First IEEE International Conference on*, pages 407 – 415, march 2003. doi: 10.1109/PERCOM.2003.1192765. 60

[Nocetti et al., 2002] F.G. Nocetti, J.S. Gonzalez, and I. Stojmenovic. Connectivity based k-hop clustering in wireless networks. In *Proceedings of the 35th Annual Hawaii International Conference on System Sciences*, pages 2450–2459. IEEE Comput. Soc, 2002. ISBN 0-7695-1435-9. doi: 10.1109/HICSS.2002.994183. URL `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=994183`. 12

[OPNET Technologies Inc. (Bethesda USA), 1986] OPNET Technologies Inc. (Bethesda USA). Opnet: Making networks and applications perform. `http://www.opnet.com/`, 1986. 46, 76, 93, 117

[Palma and Curado, 2012] David Palma and Marilia Curado. Onto scalable ad-hoc networks: Deferred routing. *Computer Communications*, 35(13):1574 – 1589, 2012. ISSN 0140-3664. doi: http://dx.doi.org/10.1016/j.comcom.

2012.04.026. URL `http://www.sciencedirect.com/science/article/pii/S014036641200151X`. 31

[Perkins et al., 2003] C. Perkins, Nokia Research Center, E. Belding-Royer, University of California, Santa Barbara, S. Das, and University of Cincinnati. Ad hoc on-demand distance vector (aodv) routing. *IETF Request For Comments 3561*, 2003. URL `http://www.ietf.org/rfc/rfc3561.txt`. 27

[Perkins and Royer, 1999] C.E. Perkins and E.M. Royer. Ad-hoc on-demand distance vector routing. In *Proceedings WMCSA'99. Second IEEE Workshop on Mobile Computing Systems and Applications*, pages 90–100. IEEE, 1999. ISBN 0-7695-0025-0. doi: 10.1109/MCSA.1999.749281. URL `http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=749281`. 27

[Perkins et al., 1994] Charles E. Perkins, T.J. Watson Research Center IBM, NY 10562 Hawthorne, Pravin Bhagwat, Computer Science Department, University of Maryland, and MD 20742 College Park. Highly dynamic destination-sequenced distance-vector routing (dsdv) for mobile computers. 1994. URL `http://www.cs.virginia.edu/~cl7v/cs851-papers/dsdv-sigcomm94.pdf`. 27

[Pfeifer and Elias] Tom Pfeifer and Dirk Elias. Commercial hybrid ir/rf local positioning system. 58

[Pomportes et al., 2010] S. Pomportes, J. Tomasik, and V. Vèque. Ad hoc network in a disaster area: A composite mobility model and its evaluation. In *Advanced Technologies for Communications (ATC), 2010 International Conference on*, pages 17–22, Oct 2010. doi: 10.1109/ATC.2010.5672729. 108

[Prasad, 2014] R. Prasad. *5G: 2020 and Beyond:*. River Publishers Series in Communications. River Publishers, 2014. ISBN 9788793237131. URL `https://books.google.pt/books?id=Et5QBAAAQBAJ`. 131

[Qiang et al., 2008] Zhang Qiang, Zhuo Ying, and Gong Zheng-hu. A trust-related and energy-concerned distributed manet clustering design. In *Intelligent System and Knowledge Engineering, 2008. ISKE 2008. 3rd International Conference on*, volume 1, pages 146 –151, nov. 2008. doi: 10.1109/ISKE.2008.4730915. 20

[Raghavan et al., 2010] A.N. Raghavan, H. Ananthapadmanaban, M.S. Sivamurugan, and B. Ravindran. Accurate mobile robot localization in indoor environments using bluetooth. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 4391 –4396, may 2010. doi: 10.1109/ROBOT.2010.5509232. 60

[Rhee et al., 2008] I. Rhee, M. Shin, S. Hong, K. Lee, and S. Chong. On the Levy-Walk Nature of Human Mobility. *2008 IEEE INFOCOM - The 27th Conference on Computer Communications*, pages 924–932, April 2008. ISSN 0743-166X. doi: 10.1109/INFOCOM.2008.145. 108

[Rodriguez et al., 2005] Miguel Rodriguez, Juan P. Pece, and Carlos J. Escudero. Inbuilding location using bluetooth. In *in In Proceedings of the International Workshop on Wireless Ad Hoc Networks*, 2005. 59

[Ros and Ruiz, 2007a] Francisco J. Ros and Pedro M. Ruiz. Cluster-based olsr extensions to reduce control overhead in mobile ad hoc networks. In *Proceedings of the 2007 International Conference on Wireless Communications and Mobile Computing*, IWCMC '07, pages 202–207, New York, NY, USA, 2007a. ACM. ISBN 978-1-59593-695-0. doi: 10.1145/1280940.1280985. URL `http://doi.acm.org/10.1145/1280940.1280985`. 95

[Ros and Ruiz, 2007b] Francisco J. Ros and Pedro M. Ruiz. Cluster-based olsr extensions to reduce control overhead in mobile ad hoc networks. In *Proceedings of the 2007 International Conference on Wireless Communications and Mobile Computing*, IWCMC '07, pages 202–207, New York, NY, USA, 2007b. ACM. ISBN 978-1-59593-695-0. doi: 10.1145/1280940.1280985. URL `http://doi.acm.org/10.1145/1280940.1280985`. 30

[Ryu et al., 2002] Jung-hee Ryu, Sanghwa Song, and Dong-Ho Cho. New clustering schemes for energy conservation in two-tiered mobile ad hoc networks. *IEEE Transactions on Vehicular Technology*, 51(6):1661–1668, November 2002. ISSN 0018-9545. doi: 10.1109/TVT.2002.804865. URL `http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=1175220`. 19

[Safa et al., 2009] Haidar Safa, Hassan Artail, and Diana Tabet. A cluster-based trust-aware routing protocol for mobile ad hoc networks. *Wireless Networks*, 16(4): 969–984, May 2009. ISSN 1022-0038. doi: 10.1007/s11276-009-0182-1. URL `http://link.springer.com/10.1007/s11276-009-0182-1`. 34

[Sanchez et al., 2009] D. Sanchez, M.A. Quintana, and J.L. Navarro. Wlan location determination using probability distributions with search area reduction via trilateration. In *Wireless and Mobile Communications, 2009. ICWMC '09. Fifth International Conference on*, pages 328 –333, aug. 2009. doi: 10.1109/ICWMC.2009.61. 59

[Sánchez and Manzoni, 2001] Miguel Sánchez and Pietro Manzoni. Anejos: a java based simulator for ad hoc networks. *Future Generation Computer Systems*, 17(5):573 – 583, 2001. ISSN 0167-739X. doi: 10.1016/S0167-739X(00)00040-6. I: Best of Websim99. II: Traffic Simulation. 107

[Sanyal and Prasad, 2014] Rajarshi Sanyal and Ramjee Prasad. Enabling cellular device to device data exchange on wisdom 5g by actuating cooperative communication based on smnat. *Int. J. Interdiscip. Telecommun. Netw.*, 6(3):37–59, July 2014. ISSN 1941-8663. doi: 10.4018/ijitn.2014070104. URL `http://dx.doi.org/10.4018/ijitn.2014070104`. 131

[Seidel and Rappaport, 1992] S.Y. Seidel and T.S. Rappaport. 914 mhz path loss prediction models for indoor wireless communications in multifloored buildings. *Antennas and Propagation, IEEE Transactions on*, 40(2):207 –217, feb 1992. ISSN 0018-926X. doi: 10.1109/8.127405. 59

[Shen, 2015] X. Shen. Device-to-device communication in 5g cellular networks. *Network, IEEE*, 29(2):2–3, March 2015. ISSN 0890-8044. doi: 10.1109/MNET.2015.7064895. 1

[Shih and Yen, 2006] Tzay-Farn Shih and Hsu-Chun Yen. Core location-aided cluster-based routing protocol for mobile ad hoc networks. pages 223–228, July 2006. URL `http://dl.acm.org/citation.cfm?id=1981726.1981768`. 32

[Smith et al., 2004] Adam Smith, Hari Balakrishnan, Michel Goraczko, and Nissanka Priyantha. Tracking moving devices with the cricket location system. In *In Proceedings of the 2nd ACM International Conference on Mobile Systems, Applications, and Services (MobiSys04*, pages 190–202. ACM Press, 2004. 62

[Stojmenovic, 2001] I. Stojmenovic. On calculating power-aware connected dominating sets for efficient routing in ad hoc wireless networks. In *International Conference on Parallel Processing, 2001.*, pages 346–354. IEEE, 2001. ISBN 0-7695-1257-7. doi: 10.1109/ICPP.2001.952080. URL `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=952080`. 18

[Tenhunen et al., 2005] J. Tenhunen, V. Typpo, and M. Jurvansuu. Stability-based multihop clustering protocol. *Personal, Indoor and Mobile Radio Communications,* *2005. PIMRC 2005. IEEE 16th International Symposium on*, 2:958–962 Vol. 2, Sept. 2005. doi: 10.1109/PIMRC.2005.1651583. 22

[Tolba et al., 2007] F.D. Tolba, D. Magoni, and P. Lorenz. A stable clustering algorithm for highly mobile ad hoc networks. *Systems and Networks Communications, 2007. ICSNC 2007. Second International Conference on*, pages 11–11, Aug. 2007. doi: 10.1109/ICSNC.2007.9. 23

[Vahdat and Becker, 2000] Amin Vahdat and David Becker. Epidemic routing for partially-connected ad hoc networks. Technical report, 2000. 90

[Venkateswaran et al., 2006] P. Venkateswaran, Rahul Ghosh, Aritra Das, Salil Kumar Sanyal, and Rabindranath Nandi. An Obstacle Based Realistic Ad-Hoc Mobility Model for Social Networks. *Journal of Networks*, 1(2):37–44, June 2006. ISSN 1796-2056. doi: 10.4304/jnw.1.2.37-44. URL `http://ojs.academypublisher.com/index.php/jnw/article/view/01023744`. 90

[Wang and Low, 2013] Ting Wang and Chor Ping Low. Evaluating inter-arrival time in general random waypoint mobility model. *Ad Hoc Networks*, 11(1):124 – 137, 2013. ISSN 1570-8705. doi: http://dx.doi.org/10.1016/j.adhoc.2012.04.011. 107

[Wang and Bao, 2007] Yu-Xuan Wang and Forrest Sheng Bao. An Entropy-Based Weighted Clustering Algorithm and Its Optimization for Ad Hoc Networks. In *Third IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob 2007)*, pages 56–56. IEEE, October 2007. ISBN 0-7695-2889-9. doi: 10.1109/WIMOB.2007.4390850. URL `http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=4390850`. 22

[Want et al., 1992] Roy Want, Andy Hopper, Veronica Falcão, and Jonathan Gibbons. The active badge location system. *ACM Trans. Inf. Syst.*, 10:91–102, January 1992. ISSN 1046-8188. doi: http://doi.acm.org/10.1145/128756.128759. 58

[Wei et al., 2007] Dali Wei, H.A. Chan, E.L. Chuwa, and B.L. Majugo. Mobility-sensitive clustering algorithm to balance power consumption for mobile ad hoc networks. *Wireless Communications, Networking and Mobile Computing, 2007. WiCom 2007. International Conference on*, pages 1645–1648, Sept. 2007. doi: 10.1109/WICOM.2007.414. 19

[Wu and Wang, 2012] Jie Wu and Yunsheng Wang. Social feature-based multi-path routing in delay tolerant networks. In *2012 Proceedings IEEE INFOCOM*, pages 1368–1376. IEEE, March 2012. ISBN 978-1-4673-0775-8. doi: 10.1109/INFCOM.

2012.6195500. URL http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=6195500. 89

[Xu et al., 2010] Hui Xu, Xianren Wu, Hamid Sadjadpour, and J.J. Garcia-Luna-Aceves. A unified analysis of routing protocols in MANETs. *IEEE Transactions on Communications*, 58(3):911–922, mar 2010. ISSN 0090-6778. doi: 10.1109/TCOMM.2010.03.080554. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5426524. 2

[Xu and Wang, 2006] Yi Xu and Wenye Wang. MEACA: Mobility and Energy Aware Clustering Algorithm for Constructing Stable MANETs. In *MILCOM 2006*, pages 1–7. IEEE, oct 2006. ISBN 1-4244-0617-X. doi: 10.1109/MILCOM.2006.302167. URL http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=4086527. 16

[Xun, 2013] Gao Xun. A mobility based clustering algorithm in MANETs utilizing Learning Automata. pages 6398–6402, 2013. URL http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=6640561. 16

[Yu et al., 2008] Jane Y. Yu, Peter H. J. Chong, and Mingyang Zhang. Performance of Efficient CBRP in Mobile Ad Hoc Networks (MANETS). In *2008 IEEE 68th Vehicular Technology Conference*, pages 1–7. IEEE, September 2008. ISBN 978-1-4244-1721-6. doi: 10.1109/VETECF.2008.18. URL http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=4656850. 33

[Yu and Chong, 2006] Jane Yang Yu and Peter Han Joo Chong. An efficient clustering scheme for large and dense mobile ad hoc networks (MANETs). *Computer Communications*, 30(1):5–16, December 2006. ISSN 01403664. doi: 10.1016/j.comcom.2006.07.014. URL http://www.sciencedirect.com/science/article/pii/S0140366406002660. 33

[Yu and Chong, 2003] J.Y. Yu and P.H.J. Chong. 3hBAC (3-hop between adjacent clusterheads): a novel non-overlapping clustering algorithm for mobile ad hoc networks. In *2003 IEEE Pacific Rim Conference on Communications Computers and Signal Processing (PACRIM 2003) (Cat. No.03CH37490)*, volume 1, pages 318–321. IEEE, 2003. ISBN 0-7803-7978-0. doi: 10.1109/PACRIM.2003.1235781. URL http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=1235781. 13

[Zadin and Fevens, 2013] Abedalmotaleb Zadin and Thomas Fevens. Maintaining Path Stability with Node Failure in Mobile Ad Hoc Networks. *Procedia Computer Science*, 19:1068–1073, 2013. ISSN 18770509. doi: 10.1016/j.procs.

2013.06.150. URL `http://www.sciencedirect.com/science/article/pii/S1877050913007588`. 7

[Zhang and Dantu, 2010] Huiqi Zhang and Ram Dantu. Predicting social ties in mobile phone networks. In *2010 IEEE International Conference on Intelligence and Security Informatics*, pages 25–30. IEEE, 2010. ISBN 978-1-4244-6444-9. doi: 10.1109/ISI.2010.5484780. URL `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5484780`. 87

[Zhang and Zhao, 2009a] Yang Zhang and Jing Zhao. Social network analysis on data diffusion in delay tolerant networks. In *Proceedings of the tenth ACM international symposium on Mobile ad hoc networking and computing - MobiHoc '09*, page 345, New York, New York, USA, May 2009a. ACM Press. ISBN 9781605585314. doi: 10.1145/1530748.1530806. URL `http://dl.acm.org/citation.cfm?id=1530748.1530806`. 89

[Zhang and Zhao, 2009b] Yang Zhang and Jing Zhao. Social network analysis on data diffusion in delay tolerant networks. In *Proceedings of the Tenth ACM International Symposium on Mobile Ad Hoc Networking and Computing*, MobiHoc '09, pages 345–346, New York, NY, USA, 2009b. ACM. ISBN 978-1-60558-624-3. doi: 10.1145/1530748.1530806. URL `http://doi.acm.org/10.1145/1530748.1530806`. 89

[Zhang et al., 2015] Zhiwei Zhang, Yunzhou Li, Kaizhi Huang, and Chen Liang. On stochastic geometry modeling of WLAN capacity with dynamic sensitive control. In *2015 13th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*, pages 78–83. IEEE, May 2015. ISBN 978-3-9018-8274-6. doi: 10.1109/WIOPT.2015.7151056. URL `http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=7151056`. 64

[Zhao et al., 2015] Chen Zhao, Mihail L. Sichitiu, and Injong Rhee. N-body: A social mobility model with support for larger populations. *Ad Hoc Networks*, 25:185–196, February 2015. ISSN 15708705. doi: 10.1016/j.adhoc.2014.10.005. URL `http://www.sciencedirect.com/science/article/pii/S1570870514002212`. 91

[Zhu et al., 2013] Ying Zhu, Bin Xu, Xinghua Shi, and Yu Wang. A Survey of Social-Based Routing in Delay Tolerant Networks: Positive and Negative Social Effects. *IEEE Communications Surveys & Tutorials*, 15(1):387–401, January 2013. ISSN 1553-877X. doi: 10.1109/SURV.2012.032612.00004. URL `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6177189`. 88