République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université de Farhat Abbas - Sétif 1 -

THESE

Présentée à la Faculté des Sciences

Département d'Informatique

Pour l'Obtention du Diplôme de

DOCTORAT EN SCIENCES

Option: Informatique

Thème

Topologies dynamiques pour la communication et la gestion des réseaux ad hoc

Présentée par

AHLEM DRIF

Soutenu le:

Devant le jury composé de:

Pr. KAMEL Nadjet	Prof. à l'UFA-Setif-1	Présidente
Pr. BOUKERRAM Abdellah	Prof.à l'Univ-Béjaïa	Rapporteur
Dr. REFOUFI Allaoua	MC-A. à l'UFA-Setif-1	Examinateur
Pr. TARI Abdelkamel	Prof. à l'Univ-Béjaïa	Examinateur
Pr. GIORDANO Silvia	Prof. à l'Univ-ISIN-SUPSI-Suisse	Examinatrice

Résumé

La problématique de l'exploitation d'un haut niveau d'abstraction des utilisateurs mobiles pour optimiser les applications sans fil est l'un des défis les plus prometteurs dans les domaines des sciences sociales computationnelles et les réseaux de communication. Notre travail de thèse propose trois contributions principales: Dans la première contribution, nous proposons une stratégie d'auto-organisation des dispositifs mobiles basées sur des topologies distribuées et efficaces en investissant l'interêt commun des utilisateurs et leurs aspects sociaux. Notre deuxième contribution analyse la corrélation entre les comportements des utilisateurs, en effet, cette dimension sociale peut être vue comme étant la motivation principale des mouvements des individus durant leur vie de tous les jours, ainsi, nous divisons notre travail en trois phases: la phase de filtrage des données mobiles, la phase de création des communautés et la phase d'analyse de modèle obtenu. Dans notre troisième contribution, nous introduisons l'apprentissage de la dynamique des communuatés mobiles. Dans cette optique, nous proposons un nouvel algorithme de prédiction en exploitant l'aspect temporel lié à l'historiques de mobilité d'un utilisateur et à l'historique des personnes qui lui sont liées. Nos contributions à la recherche sont très intéressantes et tirent profit des comportements sociaux des individus. Elles apportent plus d'efficacité en terme de temps et de précision et ouvrent la voie à des avancées dans le domaine de recherche de la mobilité humaine et les réseaux sans fil avec une approche très interdisciplinaire.

Mots clès :

Optimisation de la topologie, applications sans fil, méthodes de détection des communautés, fouille de données de trajectoires, motifs fréquents, analyse des données géospatiales, extraction de l'information, méthodes de prédiction de liens, et réseaux complexes.

Abstract

The issue of exploiting the abstraction level of mobiles users to optimize realistic applications is one of the most compelling challenge in computational social science and communication networks. The dissertation proposes three main contributions; in the first contribution, we propose a self organized policy to mobile devices based on efficient distributed topologies investigating the common interest of users and their social aspects. Our second contribution analyzes the correlation between user behaviours, indeed, we can see this social dimension as the main driver of individual movements in their daily life, thus, we divide our work in three phases: mobile data filtering phase, communities creation phase and an analysis phase of the created model. In our third contribution, we introduce the learning of the dynamics of the mobile communities. To this end, we propose a novel prediction algorithm by leveraging temporal aspect related to the past history of a user and the history of people related to him/her. Our research accomplishments are very interesting and take advantage of the social behaviors of individuals. They provide more efficiency in terms of time and accuracy and pave the way to advances in wireless networking and in the human mobility research field with a very interdisciplinary approach.

Keywords: *Topology optimization, wireless applications, community detection methods, trajectory mining, frequent pattern mining, analyzing geospatial big data, knowledge extraction, link prediction methods, and complex networks.*

Remerciements

Mon coeur est empli de reconnaissance envers mon Créateur, Qui m'a donné la force, la volonté, et l'enthousiasme pour mener à bien mon travail de thèse.

J'adresse mes plus vifs remerciements à mon directeur de thèse: le Professeur Boukerram Abdellah, qui a accepté de diriger cette thèse et qui en a supervisé le déroulement. Sa confiance et ses encouragements étaient un support infaillible pour arriver aux objectifs fixés au départ de ce travail de thèse.

Je tiens à remercier également Mr. Kheddouci Hammamache, Professeur à l'université Claude Bernard de Lyon - France, qui m'a accueillie au sein de son laboratoire et m'a amenée vers cette thématiques très riche, et m'a permis d'explorer des domaines de recherche interdisciplinaires et prometteurs.

J'adresse mes sincères remerciements au Professeur L. Louail, le Doyen de la faculté des sciences, pour avoir soutenu et encouragé toute initiative pour réussir nos travaux de recherches. Mes remercienment vont également á Dr C. Khentout, le chef de département d'informatique, pour sa disponibilité.

J'adresse mes plus vifs remerciements à Madame N. Kamel, Professeure à l'université de Sétif, d'avoir accepté de présider le jury de soutenance. J'exprime aussi tous mes remerciements à l'ensemble des membres de mon jury: Madame S. Giordano, et Messieurs Dr A. Refoufi et Prof A. Tari. Veuillez recevoir l'expression de ma profonde gratitude.

Je tiens à remercier mon collègue Slimani Yacine, avec qui j'ai travaillé sur le problème de la détection de communautés, pour avoir crée un environnement d'études convivial et persévérant durant toute cette formation. Il m'a toujours répeté: le plus important est de continuer à travailler: absolument, il n'y a que le travail qui paie.

J'exprime ma reconnaissance à la Professeure Silvia Giordano qui m'a accueillie au sein de son équipe du laboratoire "LabNetworking" de l'université des sciences appliquées, en Suisse. La Professeure Silvia m'a apporté d'enrichissantes discussions, et a contribué efficacement à certains aspects de cette thèse, à savoir l'analyse de la mobilité humaine. Son souci du détail m'a incitée à approfondir ma réflexion, sa patience à répondre à mes questions et dissiper mes doutes m'a été précieuse.

Je remercie aussi Michela Papandrea et toute l'équipe de "LabNetworking". Leur convivialité m'a fourni un cadre de travail particuliérement agréable et enrichissant.

Je tiens à remercier particulièrement le professeur Yu Zheng pour m'avoir orienté vers ses travaux de recherches et d'avoir répondu à mes emails.

J'exprime mes remerciements à mon enseignant et collègue Dr. Abdelhafidh Benaouda pour ses sincères encouragements.

Je remercie aussi Mr. Abdelkrim Bouchelaghem pour ses encouragements.

Je remercie mes chères amies Nassima, Sofia, Samira, Amel et Khadidja pour leurs encouragements. J'exprime ma gratitude á toi Sofia; tu m'as tenu compagnie avec bienveillance, tu m 'as épaulée pour réussir à aller jusqu'au bout. Merci beaucoup pour ton écoute et ton soutien infaillible.

A mes chèrs parents, mes frères, mes soeurs, mon cousin Abdelghafour et mes beaux frères, vous ne comprenez pas comment je supporte un tel rythme de travail acharné, mais vous êtes toujours soucieux de ma carrière et ouverts aux discussions parfois vives! Merci de croire en moi!.

J'éprouve une profonde reconnaissance à Ikram sur qui j'ai toujours pu compter, pendant les moments les plus difficiles. Merci d'être une soeur si merveilleuse. Tu es une source de courage infinie, chère soeur. Merci!

Une thèse est aussi une expérience humaine très riche en partages de valeurs communes, cela m'a procuré un profond contentement personnel.

Merci

...

A ceux qui agissent en bien est reservée la meilleure rècompense et même davantage . Saint Coran

Ce n'est pas en améliorant la bougie qu' on a inventé l' ampoule électrique. Le physicien Niels Bohr

Hard work beats talent when talent fails to work hard. Kevin Durant, basketball player

If learning the truth is the Sientist's goal, then he must make himself the enemy of all that he reads. Ibn alHaytham

Radiate an energy of serenity and peace so that you have an uplifting effect on those you come into contact with. Dr.Wayne Dayer

Contents

Table of contents

List of figures **iv**

List of tables **vi**

1 Introduction **1**

Introduction **1**
1.1 Problem statement . . . 1
1.2 Main issues and Contributions . . . 2
1.3 Outline . . . 7

2 Analysis of Human Mobility for Mobile Wireless Networks **9**
2.1 Introduction . . . 9
2.2 Analysis of Human mobility . . . 9
2.2.1 Definitions . . . 11
2.2.1.1 Flight . . . 11
2.2.1.2 Pause time . . . 11
2.2.1.3 Radius of gyration . . . 11
2.2.1.4 Gap . . . 12
2.2.1.5 Visit point or stay point . . . 12
2.2.1.6 Hot spot . . . 12
2.2.1.7 Jump size . . . 12
2.2.1.8 Truncated power-law distribution . . . 12
2.2.1.9 Fractal waypoints . . . 13
2.2.1.10 Inter-contact time . . . 13
2.2.1.11 Return time probability . . . 13
2.2.2 Spatial, Temporal and Social Dimensions . . . 13
2.2.3 Mobility Data Sources in Mobile Devices . . . 14
2.3 Human Mobility Features . . . 16
2.3.1 Mobility Features from a Location-Centric Perspective . . . 16
2.3.2 Mobility Features from a User-Centric Perspective . . . 17
2.4 Existing Human Mobility Models . . . 18
2.4.1 Random models . . . 19
2.4.2 Random variant models . . . 20
2.4.3 Geographical models . . . 21

2.4.4 Social models . . . 21
2.5 Conclusion . . . 23

3 Topology Optimization in Wireless Networks **24**
3.1 Introduction . . . 24
3.2 Topology optimization problem . . . 24
3.2.1 Features of communication graph . . . 25
3.2.2 Virtual topologies classification . . . 26
3.2.3 Covering node Sets . . . 27
3.2.3.1 Randomized virtual backbone construction . . . 27
3.2.3.2 Dominating and Independent sets . . . 28
3.2.3.3 Connected Dominating sets . . . 29
3.2.4 Link based structure . . . 29
3.2.4.1 Minimum Spanning Trees . . . 29
3.2.4.2 K-neighbors graph . . . 30
3.2.4.3 Relative Neighborhood Graphs (RNG) . . . 31
3.2.5 Mixed structures . . . 32
3.3 Topology Control Protocols . . . 32
3.4 A novel social perspective of topology optimization problem . . . 35
3.4.1 Virtual topology layer and routing . . . 36
3.4.2 Social layer and virtual topology layer . . . 37
3.4.3 Physical layer and virtual topology layer . . . 37
3.5 Conclusion . . . 39

4 Community Detection Methods in Complex Networks **40**
4.1 Introduction . . . 40
4.2 Communities description . . . 42
4.2.1 Graph concepts . . . 42
4.2.2 Comparative definitions . . . 42
4.2.3 Self-referring definitions . . . 43
4.2.4 Quality Functions . . . 43
4.3 Taxonomy of community discovery methods . . . 45
4.4 Agglomerative methods . . . 47
4.4.1 Algorithms based on a dynamic process . . . 48
4.4.1.1 Random walk techniques . . . 48
4.4.1.2 Synchronization techniques . . . 49
4.4.1.3 System of spins . . . 49
4.4.2 Algorithms based on modularity optimization . . . 50
4.4.2.1 Simulated annealing . . . 50
4.4.2.2 Greedy techniques . . . 50
4.4.3 Algorithms based on topological organization . . . 51
4.4.3.1 Clique . . . 51
4.4.3.2 Motif . . . 52
4.4.4 Algorithms based on spectral analysis . . . 52
4.4.5 Algorithm based on clustering properties . . . 53
4.4.6 Algorithm based on local properties . . . 54
4.4.7 Algorithm based on global properties . . . 54
4.5 Divisive methods . . . 54

4.5.1 Random walk techniques 55
4.5.2 Algorithms based on modularity optimization 55
4.5.3 Algorithms based on spectral analysis 56
4.5.4 Algorithm based on local properties 57
4.5.5 Algorithm based on global properties 58
4.5.6 Algorithm based on clustering coefficient properties 59
4.6 Dynamic community discovery methods 61
4.7 Conclusion 63

5 An Ad Hoc Topology Scheme Founded on Community Structure Proprety 64
5.1 Introduction 64
5.2 Constructing topology based on structural equivalence 65
5.2.1 Preliminary and definitions 65
5.2.2 Neighbors detection 67
5.2.3 Clustering tree Algorithm 67
5.3 On construction of virtual community backbone 68
5.4 Results 69
5.5 Routing application and topology features 74
5.5.1 Key idea 74
5.5.2 Relay region 75
5.5.3 Energy consumption model 76
5.5.4 Energy Aware Route Discovery Process 76
5.5.5 Results 78
5.6 Conclusion 79

6 A Pattern Extraction Methodology for Mobility Paths 80
6.1 Introduction 80
6.2 Proposed Methodology 82
6.3 Stay point extraction 82
6.4 Pattern discovery 86
6.4.1 Defining point of interest 86
6.4.2 Frequent patterns mining 88
6.4.3 Community discovery method 89
6.4.3.1 Fast algorithm 90
6.4.3.2 Random walk algorithm 91
6.4.3.3 Community detection Algorithm based on Relevant Coefficient 91
6.5 Experimental results 92
6.5.1 GeoLife overview 93
6.5.2 Interest Based Mobile Communities Extraction 94
6.6 Conclusion 100

7 A Novel Prediction Method for Analyzing Users Future Movement 101
7.1 Introduction 101
7.2 Problem statement 102
7.3 Data set preprocessing 103
7.4 Location Prediction based on mobile communities 104
7.4.1 User's communities pattern extraction 105
7.4.2 Community related features 106

7.4.3 Prediction Model . . . 107
7.5 Experiments . . . 108
7.5.1 Communities and mobility . . . 108
7.5.2 Prediction results . . . 111
7.6 Conclusion . . . 115

Conclusion **116**

Bibliography **119**

List of figures

2.1 Factors of influence on individual mobility behavior 10
2.2 model used to extract flight information. 11
2.3 Sample GPS traces from the Disney World scenario 15
2.4 Mobility Models 20

3.1 Virtual topologies classification 26
3.2 Application strategy for covering node sets. 27
3.3 Active connectivity and active domination of the virtual backbone 28
3.4 Dominating set (surrounded)/Maximal Independent Set (black) 28
3.5 A Connected Dominating Set (CDS) 29
3.6 Minimum spanning tree and RA 31
3.7 Relative Neighborhood Graph 31
3.8 A taxonomy of topology control techniques 33
3.9 Intuition behind the CBTC protocol. 34
3.10 Our view of Social Layer in the protocol stack. 36
3.11 The importance of appropriately setting the transmit power levels. 38

4.1 Network structure of the Web site of UAFS 41
4.2 Community discovery methods: a. Agglomerative methods. 46
4.3 Community discovery methods: a. Separative methods. 47
4.4 Community structure of Southern women-event network 47
4.5 pectral algorithm by Donetti and Muñoz 53
4.6 Schematic illustration of the edge clustering coeffcient 59
4.7 Example of a dynamic in communities 61

5.1 Structural equivalence between two nodes. 66
5.2 Initial network 72
5.3 An example illustrates the network topology 73
5.4 Comparaison between MIS backbone and virtual community backbone 74
5.5 Using a relay region 75
5.6 Network lifetime 78
5.7 Rate of received packets of DSR vs PCDSR 78

6.1 A Methodology for Human Mobility Analysis 83
6.2 Illustration of the elements of Feature Vector 88
6.3 Mobiles Paths Database 93
6.4 GPS trajectory and the extracted stay points 94
6.5 Distribution of radius of gyration 95
6.6 a)Profiling of user 126 for different activities b)Points of interest for user 126 . 95
6.7 a)Profiling of user 167 for different activities b)Points of interest for user 167 . 96

6.8 Extracted activities does user practice within different clusters 96
6.9 Frequent mining strategy to find similar users experiences. 97
6.10 The results of detecting communities using Fast algorithm 98

7.1 Link prediction problem for Interest Based Mobile Communities 103
7.2 Number of stay points per week in 2009. 104
7.3 Number of users per week in 2009 104
7.4 Users movments during 52 weeks 104
7.5 Illustration of users similarities 106
7.6 Dynamic of IMoComm over four days 110
7.7 Analyzing and excpecting the dynamic of IMoComm 112
7.8 Community type for users 'future links. 112
7.9 Illustartion of next IMoComms for user 38. 113
7.10 Performance of the prediction method for three social graphs during one month 114

List of tables

4.1 Community detection methods conducted on small graph 44
4.2 Community detection methods applied on banchmark of 1854 nodes. 44

5.1 Structural equivalence and structural weight of the studied network. 73

6.1 Characteristics of mobility dataset 93
6.2 Rate of inner community edges 99
6.3 Modularty values 99

7.1 Training and test periods for link prediction for three datasets 111
7.2 Prediction search space 115

Chapter 1

Introduction

RECENT advances in mobile communication consider the social aspects of the mobile network users to resolve various issues and to optimize the performance through the solutions of properties of complex networks. In such a mobile network where the movements of the communicating devices mirror those of their owners, finding a route between two connected or disconnected devices implies uncovering habits in human movements and patterns in their connectivity (i.e., frequencies of meetings, average duration of a contact, etc.), and exploiting them to predict future encounters. It can therefore be assumed that the future of these devices and where they are used will largely determine future innovations surrounding the proliferation of mobile devices.

1.1 Problem statement

The main motivation behind studying social computing and mobile wireless network is to improve a powerful environment for wireless and pervasive applications. Thereby, understanding human mobility patterns is a significant research endeavour that has recently received considerable attention. Moreover, individuals are set to benefit from this area of research, as mobile devices will be able to analyze their mobility pattern and offer context-aware assistance and information. However, links between mobile entities are created and broken, as the mobile entity moves within the network. Such mobility affects not only the source and/or destination but also intermediate participants in the communication, due to the networks multihops nature. The resulting routes can be extremely volatile, thus, making successful wireless access depends on efficiently reacting to these topology changes. This issue allow us to address two interesting challenges:

First, the user movements must be understood in order to improve prediction of topology changes and allow available resources to take the best decision about the dynamic of the wireless network at real-time. In addition, we consider the fact that mobility at individual level unveils interesting information: location, time and interest. To achieve this goal, we focus on analyzing the user's mobility habits and his/her future movements, and, also, extracting a correlation between users patterns to point out the social context of user's mobility. In the second issue, we manage to take advantage of the users information and his/her social patterns to provide a self organized policy to mobile devices based on efficient distributed topologies. Thus, we will be able to construct meaningful topology with respect to a higher abstract level of the mobile users in order to overcome technical barriers for mobile devices and ensure their self-organization cooperatively.

Analyzing Human mobility behaviour to build efficient topologies dedicated to several mobile applications play a major role for:

- incorporating all the information that is included in the three dimensions: locations, time and social contents,
- facilating discovery of services for mobile devices,
- improving network throughput and capacity (conserve energy of mobile devices),
- reducing the mobile data traffic,
- using fully decentralized multihop wireless communication would be possible by reinstalling the communication configuration as quickly as possible,
- repairing the disconnection in the network structure by deploying temporary communication model where each mobile entity is aware about its neighbors,
- exploiting any opportunity that brings mobile data closer to the destination.

1.2 Main issues and Contributions

Human mobility has undergone profound changes in recent decades with the development of new technologies which have transformed our daily lives. People have now sometimes very complex forms of mobility which need new vision in the analysis of dimensions that influence human mobility in order to extract relevant knowledge and predict an efficiently models. In the work [KS01], Kakihara et al. have expanded the concept of mobility by raising a number of significant aspects. It has been shown that 'being mobile' is not just a matter of people traveling but, far more importantly, related to the interaction they perform, the way in which they interact with each other in their social lives. This implies that social interactions playing crucial role as space and time aspects on studying human mobility which supplies a new perspective of mobile communication that can help us to enhance interaction among mobile users, through the protocol stack. Therefore, this thesis makes several contributions to accommodate new models in social layer and find out efficient strategies that exploit the social models in virtual topology layer and low-level layers through the protocol stack. Our contributions was a result of the understanding of many topics. The research questions of this thesis resulted in the next contributions:

- An ad hoc network is a collection of devices, which can be mobile, equipped with wireless network interfaces. The devices, also termed mobile nodes, may automatically form a temporary network among themselves without the aid of any established infrastructure or centralized control. The topology of ad-hoc networks is autonomously formed based on nodes' physical locations and transmission ranges[MH]. To understand the interaction between virtual topology layer and the routing layer, we have proposed a new energy efficient algorithm for the dynamic source routing protocol (DSR) which is based on an enclosure topology and relay region. This topology provides an energy efficient communication through the relay node and ensures connectivity between this relay node and all the existing nodes in its enclosure. This work is presented in the following academic publications:

    1. **Journal:** Ahlem Drif, Abdellah Boukerram, Energy efficient DSR Algorithm based on topology for Mobile Ad-Hoc Network, International Journal of Computer Applications, Volume 83 - Number 11, December 2013.
    2. **Conference:** Ahlem Drif, Abdellah Boukerram, Optimisation de la consommation d'énergie dans les réseaux mobiles ad hoc, Doctoriales de Sciences et technologies de l'information et de la communication STIC'11, 20-21 Avril 2011, Tébessa, Algérie.
    3. **Conference:** Ahlem Drif, Mécanisme de contrôle de topologie dans les réseaux ad hoc, in Embedded Systems Conference, EMP, Bordj-El-bahri, Alger, 05-06 Mai 2009.

- The description of the structure of complex networks has been one of the focus of attention of the scientific community in the recent years. The levels of description range from the microscopic (degree, clustering coefficient, centrality measures, etc., of individual nodes) to the macroscopic description in terms of statistical properties of the whole network (degree distribution, total clustering coefficient, degree-degree correlations, etc). The general notion of community structure in complex networks was first pointed out in the physics literature by Girvan and Newman [GN02], and refers to the fact that nodes in many real networks appear to group in subgraphs in which the density of internal connections is larger than the connections with the rest of nodes in the network. The community structure has been empirically found in many real technological, biological and social networks [EM02a], [ESMS03],[HHJ03] and its emergence seems to be at the heart of the network formation process [GDDG+03]. In our work, we aim to understand the macroscopic structure for the human mobility process in the real world. For this reason we have deeply understood the community structure, we have reviewed the existing community detection methods and have studied the network theory presented by Newman in[New10]. We have also developed two community detection algorithms. The first algorithm presents a new method for detecting communities using Markov chains based on the set of frequent motifs. The second algorithm presents an agglomerative algorithm based on relevance coefficient that keeps potentially useful information of the frequency of belonging to one community. The intuitive property is: a node has a high tendency to being members in community if it appears several times in the same community. In human mobility context, we have applied the method based on relevance coefficient for the discovery phase in our proposed methodology for human mobility analysis. Our academic publications related to communities discovery are as follows:

1. **Conference:** Ahlem Drif, Abdallah Boukerram, and Yacine Slimani. Découverte d'une structure de communauté des usagers du web. In 4 me conférence sur les modéles et l'analyse des réseaux: approches mathématiques et informatiques, Saint-Etienne, France, 2013.
2. **Journal:** Ahlem Drif et Abdellah Boukerram. Taxonomy and survey of community discovery methods in complex networks. International Journal of Computer Science and Engineering Survey, vol.5, no 4,2014.

3. **Journal:** Slimani, Y., Moussaoui, A., Lechevallier, Y., and Drif, A. (2012, January). Identification de communautés d'usage du web depuis un graphe issu des fichiers d'accés. In EGC (pp. 525-530).
4. **Journal:** Yacine Slimani, Ahlem Drif, Abdelouahab Moussaoui, Discovery and Analysis of Usage Patterns for Web Personalization. International Journal on Recent and Innovation Trends in Computing and Communication.
   DOI: 10.17762/ijritcc2321-8169.150232 pp 578-582,V(3), Issue(2), 2015.

5. **Technical Rapport:** Drif, A., Boukerram, A., Slimani, Y., Moussaoui, A. (2016). Découverte de communautés dans les réseaux complexes. Hal.archives-01389844.

6. **Journal:** Yacine Slimani, Abdelouahab Moussaoui, Yves Lechevallier, Ahlem Drif, Discovering Communities for Web Usage Mining Systems, will appear in International Journal of Advanced Intelligence Paradigms (2017).

- Exploring and analyzing human movements that are changing over time, relating to user habits and depending on spacial aspect is a challenging problem [BCH+13], [RSH+11a]. The very first step in understanding human mobility consists in having data representing such mobility in order to extract characteristics from them. Among the available mobile data sets , we have chosen the very large dataset collected in GeoLife project and released by Microsoft Research Asia [ZZXM] [Zhe] which we will present in chapter 6. Geolife dataset provides a good context information, including time, history, real location coordinates, daily activities of individuals which allows us to:

  – assess the mobility features.
  – design algorithms that can effectively learn from dynamic of mobile users.

  In this contribution, we are interested in discovering communities of mobile users and studying how do communities provide accurate knowledge to analysis the different forms of human mobility. The basic idea of our contribution is to model the behaviour of users with strong social characteristics regarding the context of their location histories to explore a similar interest of people by mining their mobiles communities. To that purpose, we have extracted, first, the stay points, each of which denotes a geographic region where an individual stayed for certain duration, from the trajectory data, and second, we have explored the semantic meaning of a physical location using a simple model to express the relationship between the locations and the activities. After that, we have designed an algorithm to mine the travel sequences between two different individuals with respect to their location histories. Then, we have derived the similar interests among people which induce the extraction of their mobile communities patterns. The proposed methodology illustrates in what way a common interest of a group of individuals can create better understanding of human mobility. Realistic models based on these interest based communities can be the basis for several applications as recommendation system or wireless networks management. The academic publication of this work is:

  1. **Journal:** Ahlem Drif, Abdellah Boukerram, Yacine Slimani, Silvia Giordano, Discovering Interest Based Mobile Communities, Mobile Networks and Applications, February 2017. DOI:10.1007/s11036-017-0811-3.

- A very significant characteristic of ad hoc networks is the presence of community structure which can be represented by a group of mobile devices that carried by humans that share common interest. The communication of such devices is, then, necessarily based on socialization behaviour, so a community organization is paramount and a number of social and technical barriers must be overcome in order for mobile ad hoc communities to self-organize cooperatively[H.R02]. Thus, we have proposed a social-aware approach for constructing efficient topology based on a common interest of mobile users. In the first phase approach, we construct a clustering tree based on structural weight of nodes with respect to the range assignment mechanism. The key idea of topology control techniques is to adjust nodes' transmission power to achieve several objectives such as reducing energy consumption, reducing interference, and increasing network capacity, while maintaining network connectivity. In the second phase, the approach ensures that a mobile can reach all members of its community based on an efficient virtual backbone of community. The comparison with Minimum Independent Dominating Set topology shows that the proposed method build a meaningful network topology. Our academic papers related to this contribution have been presented and published in:

1. **EditedProcedding:** Ahlem Drif, Abdellah Boukerram, and Yacine Slimani. Community discovery topology construction for ad hoc networks, in Lecture Notes (LNICST) Wireless Internet, pages 197-208, Springer 2014.
2. **Conference:** Ahlem Drif, Abdallah Boukerram, and Yacine Slimani. Community discovery topology construction for ad hoc networks. In International Wireless Internet Conference-The 8th International Wireless Internet Conference - Symposium on Wireless and Vehicular Communication, November 13-14, 2014, Lisbon, Portugal.

- Mobile movement prediction is defined as the prediction of a mobile user's next movement in a networked environment. In this work, we have proposed a novel prediction algorithm by leveraging temporal information on network dynamics. We have took advantage of the temporal graphs, the past history of an individual and the past history of person related to him. Thus, the proposed approach improves effectively prediction of users mobile behavior with regard to his/her next Interest based Mobile Community. Our method faithfully captures the user-specific mobility aspects and allows to deploy an efficient human mobility models that combine multiple information sources from users correlation and their behavioral patterns. Our contribution has been presented and published in:

1. **Book Chapter** Drif, Ahlem., Boukerram, Abdellah., Slimani, Yacine., and Giordano, Silvia. Can we recognize the next user's mobile community?. in Journal of Complex Networks and Their Applications. November 2017 (will appear) $DOI : 10.1007/978 - 3 - 319 - 50901 - 3_27$, Vol 5, pp.335-346.
2. **Conference** Drif, Ahlem., Boukerram, Abdellah., Slimani, Yacine., and Giordano, Silvia. Can we recognize the next user's mobile community?. in Proceedings of the

5th International Workshop on Complex Networks and their Applications, 28-30 November 2016, Milan, Italy.

1.3 Outline

The previous objectives and contributions are addressed along this dissertation following the structure described next.
Chapter 2 provides the basics and the background of human mobility analysis. It gives explanation of the characteristics of human mobility that we try to fulfill in this work. It is followed by descriptions for the terms that are frequently used in human mobility analysis and in our work. We also explain the characteristics of mobility data source in mobile devices. The chapter provides a brief overview of recent works that study the daily life mobility patterns and presents the classical human mobility models.

Chapter 3 provides an overview of the state of the art concerning the topology optimization problem. It gives a definition of topology optimization and describes the features of the communication graph. It also clarifies the basic idea of several virtual topologies and gives their classification. After having described a taxonomy of the topology control approaches, the chapter ends with a discussion on how social models can be integrated into the network protocol stack. On this point, we introduce our social perspective that express the need to deign novel social-aware approaches that take as input certain spatial, temporal and social parameters to provide valuable scheme to the topological and physical layers.

Chapter 4 contains some examples of real networks with community structure. In this way, we shall see what communities look like and why they are important: social networks are paradigmatic examples of graphs with communities, and, the word community itself refers to a social context. Then, we describe community local definitions and community comparing definitions, and, we review the notions of partition quality. Driven by the growing popularity of community detection methods in complex networks, we point out the most cited works in the literature, and, provide a taxonomy for community detection methods. Finally, chapter 4 gives a brief discussion of the issue of communities in dynamic networks.

In chapter 5, we introduce our proposed model for constructing meaningful topology based on the common interest of the users of ad hoc network and we give definitions of the structural equivalence measure. Then, we explain the process of constructing a privileged set which generally aims to get a better topology control strategy. We discuss the design of a context-aware algorithm including the common interest information that characterize a mobile entity. This chapter gives a comparison with the two implemented topologies which demonstrates the efficiency of the proposed scheme. Finally, it follows by presentation of the idea key of interaction between topology layer and routing application in order to highlight the benefits of using a self organized policy to mobile devices.

Chapter 6 is devoted to explain the proposed mobility data mining methodology. Here, our ultimate goal is to extract relevant knowledge and provide an abstract level for understanding people mobility. So, this chapter describes our work in detail. The experimental results demonstrate the validity of the proposed methodology to extract what we have called "Interest based Mobile Communities" (IMoComm) in a real case.

In Chapter 7, we develop a new technique to improve users movement prediction using insights from analysis of network dynamics. The chapter explains the analysis of the dynamic of individuals at community level over different timing and thus defines communities prediction features. After performing an unsupervised task to extract the link pattern between people that distinguish meaningful Interest Based Mobile Community structures, we take advantage of temporal graphs, in a given learning period, and , therefore, we have investigated user's patterns to infer which new mobile community a user is likely to be at in the near future which result in reducing dramatically prediction space.

Finally, we conclude the thesis and introduces possible future directions.

Chapter 2

Analysis of Human Mobility for Mobile Wireless Networks

2.1 Introduction

Nowadays, the huge worldwide mobile-phone penetration is increasingly turning the mobile network into a gigantic ubiquitous sensing platform, enabling large-scale analysis and applications. With the current advent of mobile devices such as smart-phones and tablets, understanding and modeling human interactions with respect to the spacial, temporal and social dimensions can help to develop mobile infrastructure placement, transfer scheduling algorithms, mobile advertising, mobile social networking, location based services and so on. In this chapter, we point out the basic concepts and discuss the key ideas and the problems that motivate our work.

2.2 Analysis of Human mobility

Human mobility has a significant influence on the performance of systems that involve daily human activities. Therefore, studying and finding fundamental characteristics of human mobility and developing realistic human mobility models are essential for optimum construction of these systems. Recently, for achieving more reliable models, researchers have tried to use real human traces in their studies such as mobile-phone-location traces [GHB], GPS traces [ZLC+], or Wireless Data [RSH+09].

Marcel Hunecke [Hun06], have defined two different set of factors of influence on individ-

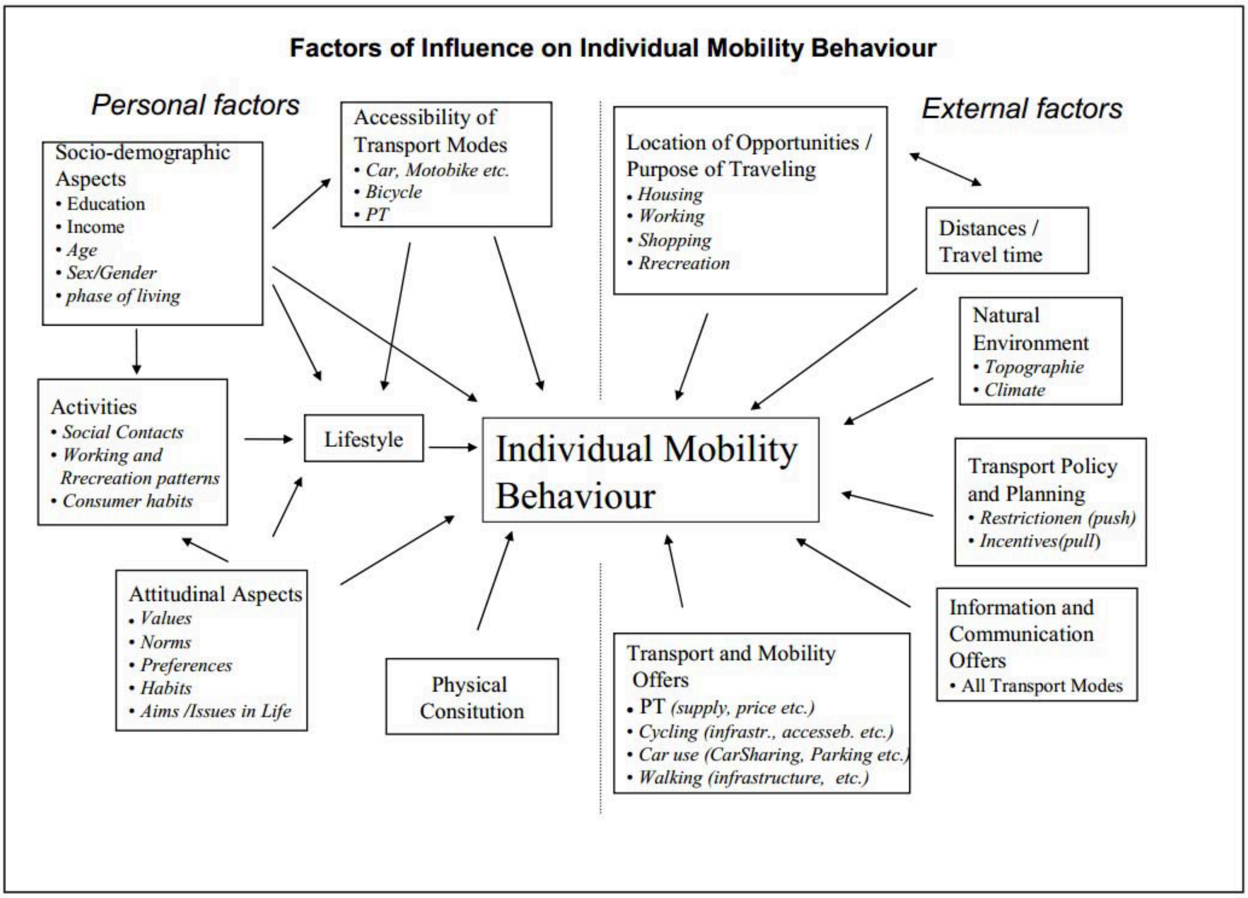


Figure 2.1: Factors of influence on individual mobility behavior[Hun06]

ual mobility behavior: Personal factors and External factors (see figure 2.1). Among these factors, two types of personal factors are mentioned as relevant factors for individual mobility: socio-demographic characteristics and attitudinal factors, which determine individual options and needs for mobility activities. Attitudinal factors include values, norms and attitudes (e.g. symbolical estimations of transport modes), which affect preferences and habits for specific activities, destinations, routes and means of transport. Furthermore, recent works try to define and extract behavioral features of individuals from individual mobility traces (such as popular places, the media and content they have accessed via internet, working pattern, ..., etc.)

We introduce,here, the following definitions and characteristics of human mobility:

2.2.1 Definitions

2.2.1.1 Flight

Flight is a straight line trip from one location to another without a pause or directional change [RSH+11b]. Gonzalez et al. [GHB] analyzed cellular network records and extracted flight lengths with the concept of flight length as the distance between tower locations which handle consecutive calls of a user. They use two data sets for exploring the mobility pattern in human beings. The first one consists of traces of 100000 individuals that were selected from six million mobilephone users recorded over six months. The second one contains locations of 206 mobilephone users recorded every two hours for one week. In all the data sets they show that the flight length follows truncated power-law distributions (see figure 2.2).

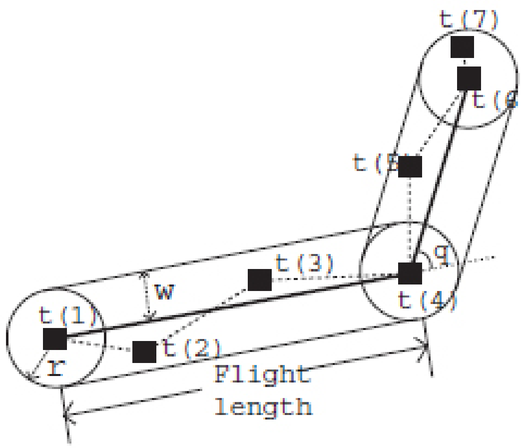


Figure 2.2: The rectangular model used to extract flight information from traces.

2.2.1.2 Pause time

Pause time is the time during which a user stays at one point before starting a new flight.

2.2.1.3 Radius of gyration

Radius of gyration is the characteristic of a users trajectory during an observation time. This parameter is used to measure how far and how frequently a user moves. Thus, it was shown that the distribution of the radius of gyration can be approximated by a truncated power law. This suggests that the majority of people usually travel in close vicinity to their home location, while few of them frequently make long journeys [SQBB].

2.2.1.4 Gap

Gap is the distance between two time-consecutive points of a user's location traces.

2.2.1.5 Visit point or stay point

Visit-point is defined as a place where a user makes a stop or pause. In [LZX+08], the stay point is defined as the location where a user stays more than 30 seconds within a circle of 5 meter radius of that location for the collected GPS traces.

2.2.1.6 Hot spot

A cluster of visit-points that are connected to each other meaning that they are within a predefined radio range.

2.2.1.7 Jump size

Jump size or travel distance characterizes the spatial dimension of users. Thus, the average travel distance can be defined by $\overline{\Delta r(l)} = \sum_{i=2}^{n} |r_i - r_{i-1}|$ where $r_l = r_1, r_2, ...r_n$ be the sequence of n geographical displacements where user l have traveled during a time period and $|r_i - r_{i-1}|$ is the distance between locations r_i and r_{i-1}. Travel distance can range from a few to thousands of kilometers over a periods of time [PWJX14].

2.2.1.8 Truncated power-law distribution

It is shown that the lengths of human flights which are defined to be straight line trips without directional change or pause (or lines between two consecutive stay points) have a truncated power-law distribution [CSN09].

2.2.1.9 Fractal waypoints

The stay points of humans can be modeled by fractal points. This implies that people are always more attracted to more popular places. The term fractal-points means the more popular points are clustered together while the less popular point are far from others. The high popular places are rare and the less popular places are plenty. The distribution of the points is the same in every level of resolution, in other words, fractal-points exhibit self-similarity [RLH+08].

2.2.1.10 Inter-contact time

Inter-contact time is defined as the time elapsed between two successive contacts of the same devices [S+06].

2.2.1.11 Return time probability

Return time probability distribution is the probability of returning at time t to a selected place. Prominent peaks (at 24, 48, 72, . . . , h) capture the tendency of humans to return regularly (on a daily basis) to a location they visited before [GHB08a].

2.2.2 Spatial, Temporal and Social Dimensions

Recent works have focus on expanding the concept of mobility by looking at three distinct dimensions of human interaction; namely, spatial, temporal and contextual mobility. The spatial features refer to the movement of people in physical space but also the global flux of objects, symbols, and mobility of space in virtual communities, temporal aspects are related to duration, sequence, temporal location and rate of recurrence of events [Bar88] whereas the social dimension is related to relational aspects of human interaction and contexts in which the human actions occur [LS00]. Social dimension of human interaction is increasingly mobilized by the impacts of various mobile technologies and the growth of social networks. Our society is transforming itself into a mobile society, where interaction itself is mobilized [SMK].

2.2.3 Mobility Data Sources in Mobile Devices

The very first step in understanding human mobility consists in having data representing such mobility in order to extract characteristics from them. Therefore, it is important that the data reflects as faithfully as possible the information related to mobility. It can be inspected from many different perspectives depending on the application.

- **Mobility levels:** general (all the movements of the user), medium small scale (mobility inside a campus), or indoor (inside a building, floor, room, etc.), among others.
- **Accuracy:** might be key (for indicating a friend where we are, or to locate ourselves in a foreign city), or can be a relaxed requirement (when looking for restaurants nearby).
- **Locating:** the user might be enough (for applications with just the current location attached to each request), or the user might need to be tracked. In this last case, the tracking frequency is also another parameter to analyze.

Nowadays almost every person carries a mobile device with her, all day long, everywhere she goes. Thus, the mobile device is the perfect proxy to track user's mobility. This fact was rapidly acknowledge by the research community, who took advantage of the increasing number of sensors as well as the available application programming interfaces (APIs) of the different operating systems, which incredibly ease the use of the systems and sensors in the devices, to develop applications capable of tracking the user's movements. Among the several systems integrated in the mobile devices that can be leveraged as location proxies, the most popular ones are GPS, as it is the only one providing real location coordinates, and also Wi-Fi and the cellular telephony network. There are other systems, like Bluetooth and Radio-frequency IDentification (RFID), which are also used in some specific scenarios to locate people in small size environments. We present the technologies available in mobile devices to track their owners location below:

1. **Global Positioning System (GPS):** The great majority of new mobile devices integrate a GPS system. When a user wants to obtain her location using this technology, first she/he enables the GPS of the terminal, then the device searches for the satellites and synchronizes with them, and finally, once the synchronization is set, the user can perform location requests. The location data accuracy is the main strength of this technology, since it is close to 10 meters [DR01] [Zha02]. However, the decrease of the battery consumption comes at the cost of reducing the accuracy of the location data obtained [CGS+09]. Figure 2.3 shows a sample of GPS traces tracked from the Disney World.

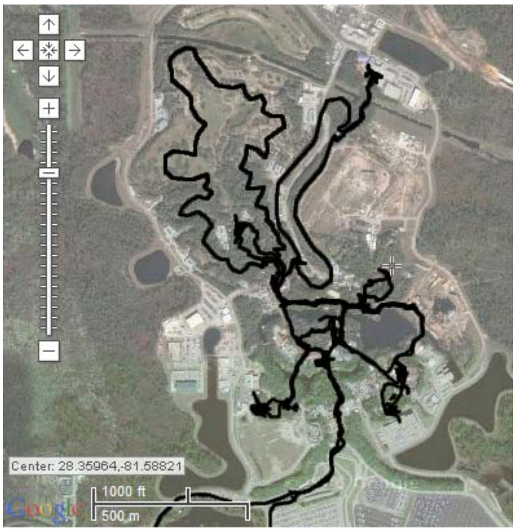


Figure 2.3: Sample GPS traces from the Disney World scenario.[RSH+11b]

2. **Wi-Fi-based location:** By monitoring the Wi-Fi access point (AP) the user's device is attached to ,as he/she moves, his/her mobility patterns can be indirectly tracked. The mapping between the AP medium access control (MAC) address and its location is needed in order to know the zone where the user is at all times. Since location tracking using Wi-Fi only needs to know the AP MAC address (no need for data transferring), the power consumption of having the Wi-Fi antenna working, scanning the radio environment looking for new Wi-Fi networks, and being attached to some AP is low [FN01] . However, depending on the method for translating the MAC address of the AP to the corresponding coordinates, there may exist an extra power consumption if an Internet connection is needed.

3. **Cellular telephony network-based location:** The working principles of this system are very similar to those of Wi-Fi case. The user mobility is tracked by knowing the network base transceiver station (BTS), also referred to as cell, the device is attached to as the user moves. In this case, a translation from BTS (cell) information to coordinates is also needed. The coverage is the main advantage of this technology since it provides global coverage, even in indoor environments. However, a cell from Global System for Mobile Communications (GSM) network ranges from 200 meters radius in urban areas to up to several kilometers in rural scenarios, thus the accuracy being much worse than GPS or Wi-Fi systems.

4. **Bluetooth and Radio-frequency IDentification (RFID):** Bluetooth-enabled devices are prevalent and the device population is relatively homogeneous. There were a several experiences collecting traces of Bluetooth activity in different urban environment and in some controlled setting [SCM+06].

5. **Location-Based Social Networks (LBSN):** This new type of social network is based on each of its users indicating (check-in) the place (restaurant, airport, sport center. . .

) where she/he is at every moment, like in Foursquare. Therefore, the location history can be directly obtained by taking the sequence of check-ins made by the user. The accuracy depends on the honesty of the user: if she/he checks-in where she/he really is, then she/he will be located inside the place she/he says to be, and depending on the size of the place, the accuracy will be higher (if the place is small, like a restaurant), or lower (if the place is big, like a mall).

2.3 Human Mobility Features

Researchers studying daily life mobility patterns have recently shown that humans are typically highly predictable in their movements [PZG+13]. The studies surveyed can be roughly classified, mainly, into two groups, depending on the perspective form which mobility is analyzed. On the one hand, many works hold a perspective centered on the environment where mobility is studied, generally at city-level. Its main goal is to uncover how mobility shapes the environment or vice versa, but with the central aspect being mobility in the environment. The opposite perspective is user-centric, that is to say, the goal in this case is to characterize the individual or group behavior, but having the person as the center piece instead of the environment.

2.3.1 Mobility Features from a Location-Centric Perspective

Considering the case of location-centered mobility, sometimes known as urban dynamics, many works have appeared in the last years in line with the advent of smart cities and the corresponding need to understand the inhabitants flows defining the scenario [STN15] to help in planning and provision of municipal facilities and services, provide better public transportation [BCDL+13] and road usage [WHB+12]. This type of studies were not possible until traditional approaches, like surveys, were replaced by the data provided by cellular network operators, which disclose more than just snapshots of people movements, but where the spatial extent and temporal correlations are wider than the ones provided by previous studies.
One of the first works using cellular telephony data to characterize urban-related features was carried on by the MIT in collaboration with Telecom Italia [RCSR07]. The study divided Rome into pixels and chose six different locations of one pixel each. The Erlang daily traffic distributions were studied, to group them by degree of similarity and map them to hot and cold areas of activity along the day, week and month. This work was taken as reference by Sun et al. [SYW+11], who also divided a southern city in China into pixels, and by using the call-data-records (CDRs) collected in such city, analyzed the population distribution based on

the cellphone usage data from the CDRs in each pixel.
More recent works, like [BCH+11], use CDRs to capture the city dynamics by determining the residential areas where people work and the residential areas of late-night people, thus demonstrating that clustering people based on cellphone usage is possible, even without taking into account temporal correlations. Follow up works [IBC+10], [IBC+11a], [IBC+11b] extended the former study by finding mobility patterns in New York and Los Angeles regions, such as identifying important locations, who travels further, who travels more distances, when people move more and at what season, among others, using a metric called daily range, which corresponds to the maximum distance traveled in a single day.
One of the most common studies regarding cities is to uncover which are the different regions of the city. In the work [YZX12], the authors try to complement the knowledge of points of interest (POIs) of a city with the information provided by CDRs to differentiate the intensity of each function in each region or location (e.g., a small restaurant has a different impact than a big attracting one, even when the two of them are considered POIs).

2.3.2 Mobility Features from a User-Centric Perspective

The second standpoint of mobility is user-centric, referring to the works aiming at characterize the intrinsic features of human mobility, disregarding the specific scenario where they move. In the survey elaborated by Lin et al. [LH14], they review relevant results in some of the main areas studied in human mobility studies: inferring important locations, detecting modes of transport, mining trajectory patterns, and recognizing location-based activities. They also classify mobility analysis into two main areas: mining mobility patterns and constructing mobility models. These two big blocks were also pointed out in [KBCP11], authors have proposed a framework that includes data collection and the final applications where the mobility results and models are applied as additional areas conforming the big picture of human mobility study. Regarding the identification of salient locations, Eagle et al. [ECQ09] analyzed CDRs of 215 individuals recorded during 5 months to cluster the most used BTSs (i.e., locations) and validated the results using data coming from Bluetooth beacons placed in the individuals homes. In [ZZXM], the authors use GPS data from 107 users taken during a year to mine interesting locations, as well as classical travel sequence, for travel recommendation.
Boldrini et al. [BP10] identify the preference to spend time in a limited number of popular locations, and study the preference to select short distances over longer ones, and compute the sociability of users. Thus, they have proposed a mobility model based on these three properties for reproducing accurately the behaviors of users in mobile ad-hoc networks (MANETs), opportunistic and delay tolerant networks.
In the works [GHB08b] [TdMGP15], the authors use the CDRs of 100,000 users collected during 6 months to study the basic laws driving human motion. They found out that human trajectories show a high degree of temporal and spatial regularity, since each individual is

characterized by time-independent travel distance and a significant probability to return to a few highly frequented locations. The distribution of the distance covered in the displacements suggests that human motion follows a Levy Walk, and the calculation of the radius of gyration (i.e., the distance traveled by the user when observed up to time t) follows a truncated Levy Flight distribution. These results were backed up by the work of Rhee et al. [RSH+11a], where GPS data were used instead of CDRs, but leading to the same heavy-tailed distribution of the distances covered by the individuals in their displacements. These behaviors, widely detected in several different data sets and populations, is in line with the exploration and preferential return model proposed in [SKWB10]. A novel finding is the two types of individuals found in [PSR+15], where the authors split the population into the so called returners (people who only visits a very limited set of locations) and explorers (people who travels to many more different and distant locations than the most usual ones), and explain the role of both types of individuals in the spread of diseases and social networking. Song et al. went a step further by proposing a new metric for mobility [SQBB10]. In their study, they used CDR data of 50.000 individuals recorded during 3 months, and study their entropy and entropy rates, to finally propose a new metric, the predictability, which sets the upper bound on the best accuracy a location prediction algorithm could ever achieve, depending on the specific user entropy rate and different number of locations visited. They show that predictability is largely independent from the radius of gyration, and that the average predictability is centered in the 93% of correct predictions. After its proposal, predictability has been widely studied.

2.4 Existing Human Mobility Models

Human mobility models can help in predicting the movements of humans. In this way they can be used in studying the effect of the topology design for finding ways to reduce the unnecessary communication signals. The optimizations also have great impacts on heterogeneous networks because of extensive communication requirement due to cooperation among varying communication network technologies. Moreover, human mobility models can be very helpful in a variety of social issues [?], for example, urban planning, traffic engineering, understanding spread pattern of diseases, traffic congestion detection (e.g. accidents), and disasters discovery. Many human mobility models in the past two decades have been developed to represent human mobility patterns. However, very few of them were validated against large-scale and detailed human traces because of many technical and legal problems. Nowdays, some attempts have been made to collect real human traces, such as mobile-phone-location traces [GHB], GPS traces [RSH+11a], [RLH+08], or trace of wireless contacts and social connections [MDX12]. There have been a lot of efforts to introduce a synthetic human mobility model that can capture human travel behavior in a realistic way. These existing models can be categorized into: random models, random variant models, geographical models, and social behavior models.

However, with all the new data captured by mobile phones about user mobility, it became clear that classical mobility models, such as random walk or random waypoint, among others, fall short to capture the real features driving human mobility. Thus, this huge amount of location data captured using mobile devices as monitoring tool needs to be carefully considered to determine which mobility-related information is able to provide, and their limitations, so that more accurate conclusions about mobility can be derived from it.
In the following paragraphs, we discuss each category of models and their ideas:

2.4.1 Random models

In random models, stay points are chosen randomly based on some probability distributions [RSH+11b].

- **Random walk model or Brownian motion model:** In this model, speeds and directions are randomly assigned to mobile nodes to select their next destination, i.e. each mobile node's speed is chosen uniformly from a defined range $[speedmin, speedmax]$, while its direction is chosen from the range $[0{-}2\pi]$. The mobile nodes move for a distance, d, or time interval, t, then it is considered that they have reached their destination.
- **Random waypoint:** This model is like the random walk except that it considers the pause time as well. When the node reaches its destination it remains there for a predefined amount of time and then selects a new destination according to a uniform distribution over the area.
- **Truncated Levy-walk model:** The Levy walk model takes after the random walk, but it represents the heavy-tail flight feature. In this model, a step is defined by flight length, direction, flight time and dwell time. Mobile nodes choose their direction randomly, then choose flight length and pause time to follow truncated power-law distributions. Flight lengths have a probability density function as follows:

  $$p(l) \sim \begin{cases} l^{-(1+\alpha)} & ; l <= l_{max} \\ 0 & ; l > l_{max} \end{cases}$$

  Where a walker makes a flight followed by a pause,θ is the direction of that flight, $l > 0$ is the length of the flight, Δt_p is the time duration of the pause or pause time, α is a constant parameter of the distribution known as the exponent or scaling parameter, and β is the displacement exponent.

  Similarly, pause times have a probability density function as follows:

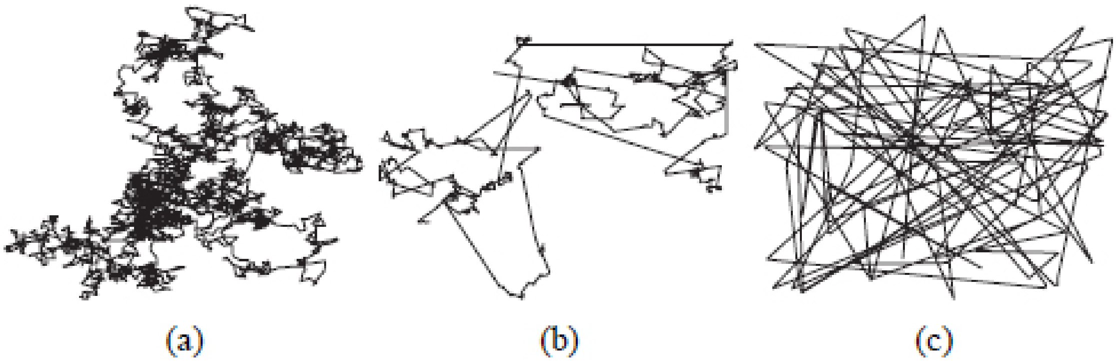


Figure 2.4: Sample trajectories of (a) Brownian motion, (b) Levy walk
and (c) Random way point

$$p(\Delta t_p) \sim \begin{cases} \Delta t_p^{-(1+\beta)} & ; \Delta t_p <= \Delta t_{p_{max}} \\ 0 & ; \Delta t_p > \Delta t_{p_{max}} \end{cases}$$

Rhee et al [RSH+11b] used mobility track logs obtained from 44 participants carrying GPS receivers from September 2006 to January 2007. The sample settings where traces are obtained are two university campuses (one in Asia and one in the US), one metropolitan area (New York city), one State fair and one theme park (Disney World). The participants walk most of times in these locations and may also occasionally travel by bus, trolley, cars, or subway trains. They have illustrated that many statistical features of human walks follow truncated power-law, showing evidence of scale-freedom. Furthermore, they have construct a simple Levy walk mobility model which is versatile enough in emulating diverse statistical patterns of human walks observed in the studied traces (see figure 2.4).

2.4.2 Random variant models

In this type, the mobility models are also random but they contain dependencies (spatial or temporal) .

- **Markovian way point model:** This model is a mobility model based on the basic random waypoint model. It implements some Markovian transition probabilities among waypoints. In other words, choosing next waypoints depends on current locations. This will create spatial dependency for the model [Meg10].
- **Gauss Markov model:** This model contains temporal dependency. The idea is that mobile nodes choose their speed and direction randomly as in random models, but after

a fixed time interval, the speed and direction are calculated again considering previously calculated values of these parameters [GHB08b].

- **Reference point group mobility model:** In this model, mobile nodes make a group and each group has a leader and the nodes move along with their leader, with the same direction and speed. That leader is considered as the reference point for the group and around that each mobile node is moving in its own way. Both the group and individual movements are based on random waypoint models. The fact that in this model each node's speed may depend on its neighbor's speed creates spatial dependency [Meg10].

2.4.3 Geographical models

These models contain geographical constraints.

- **Freeway model:** Each mobile node is limited to its lane on the freeway and also the speed of the mobile node temporally relates to its previous speed.
- **Manhattan model:** It has the same characteristics as the freeway model but the mobile nodes can make turns at each corner of the street [BSH03].
- **Obstacle model:** In order to represent realistic geographical limitations, this model introduces obstacles in pathways. These obstacles are randomly placed in the simulated area. Mobile nodes should change their paths and choose proper ways to avoid running into those obstacles [PBDK12].

2.4.4 Social models

This type of mobility models considers human mobility based on collective human behaviors which are affected by social factors such as friendships.

- **Dartmouth model:** In this model, mobile nodes are modeled to move among hot spots. Mobility information, which contains hot spots locations, transition probabilities for moving between hot spots, and pause time distribution, is extracted from real data sets. The model estimates the locations and paths of mobile nodes based on the extracted, region dependent mobility information. So, it needs to have the transition probabilities for moving between hotspots and the locations of hotspots as the input [LHK+09].

- **Clustered mobility model:** This model is based on preferential attachment theory which means the attractiveness of one area is determined by the current number of nodes that are assigned to that area. The mobile nodes tend to visit attractive areas. The result of this fact is that, areas which have high attractiveness will gain more attractiveness. This model divides the simulation area into a number of subareas and then assigns the mobile nodes to these subareas using the referred theory. Mobile nodes select their next subarea according to its attractiveness which is proportional to $(k+1)^{\alpha}$, where k is the number of nodes in the subarea and α is the clustering exponent. Consequently, the attractiveness follows power-law distribution [NSRD11].

- **ORBIT model:** In the ORBIT model the total network area is divided into a number of clusters and each mobile node is assigned a subset of these clusters. The mobile nodes are able to move randomly only within their clusters set [GPQ07].

- **SLAW: Self-similar Least Action Walk model:** In [LHK$^+$09], Lee et al. have introduced a model called SLAW which generates fractal waypoints, and uses it with the Least Action Trip Planning (LATP) algorithm to simulate human traces. The SLAW model starts with generating a map with fractal waypoints. In this model, the mobile nodes first select a subset of waypoints in the generated map. Then, the order, in which those selected waypoints are going to be visited, is specified as the mobile node's daily plan. SLAW first builds clusters of waypoints, for representing hot spots, by connecting every pair of waypoints whose distances between each other are less than 100 meters (typical Wi-Fi outdoor transmission range). Each cluster is assigned a weight which is the ratio of the cluster's size (number of waypoints in that cluster) to total number of waypoints in the map. Each mobile node chooses 3 to 5 clusters randomly from clusters set with probability proportional to these weights. In other words, the model represents cluster's popularity proportional to cluster's size. Then, each mobile node selects 5% to 10% of waypoints in each chosen cluster uniformly. To sum up, this waypoint selection algorithm creates heterogeneously bounded mobility areas, one of the characteristics of human mobility.

- **SMOOTH model:** In [MCN11], Munjal et al. have proposed a model that captures some of the features: Human-Mobility characteristics; flight lengths and pause time follow truncated power-law distributions. The idea of the model is described in two parts: visit-point placements and movement patterns. The waypoints is regrouped in clusters and their popularities are defined by randomly assigned probabilities to have the sum of all clusters' probability as one. Each cluster is represented by a single coordinate called landmark. Landmarks are placed uniformly over the simulation area such that no two landmarks are within each other's transmission range. There is no boundary defined for any cluster. For initial placement of a mobile node in the network, a mobile node selects a cluster by its probability and is placed within half of the cluster's transmission range. For the movement pattern, each mobile node chooses to explore a new location with the probability proportional to the number of distinct locations visited so far. For the new location first the flight length is generated using a power-law distribution. Based on this length and the current location of the mobile node the destination is calculated. If the node chooses to visit one of the locations it has visited before, the location is selected with probability proportional to the total number of times the node has visited the location so far.

2.5 Conclusion

This chapter provides an overview of the state of the art concerning the topic covered in this thesis. It includes some background on one of the main building blocks in which the thesis is based on; the human mobility features, mobility data source and keys concepts for further chapters.

Chapter 3

Topology Optimization in Wireless Networks

3.1 Introduction

An important problem appearing in computer-communication network is to design an optimal topology for increasing the network traffic carrying capacity. This chapter considers the case of mobile networks, and it discusses the implications of node mobility on the characterization of an optimal topology with respect to the critical range for connectivity. In the end of the chapter, we introduce our social perspective of topology optimization and discuss the features of interaction between social layer and virtual topology layer in the protocol stack.

3.2 Topology optimization problem

Definition 1 *We consider a wireless ad hoc network consisting of a set V of N wireless nodes distributed in a two-dimensional plane. By proper scaling, we assume that all nodes have the maximum transmission range equal to one unit. These wireless nodes define a unit disk graph $UDG(V)$ in which there is an edge between two nodes if and only if their Euclidean distance is at most one. Unit Disk Graphs (UDG) are the most prominent class of graphs used in wireless network [BCH+13].*

Topology Control is the art of coordinating nodes' decisions regarding their transmitting ranges, in order to generate a network with the desired properties (e.g.connectivity) while reducing node energy consumption and/or increasing network capacity.
Given the set N of network nodes, a range assignment for N is a function RA that assigns to

every $u \in N$ a transmitting range $RA(u)$, with $0 < RA(u) < r_{max}$, where r_{max} is the maximum transmitting range. Note that, under the assumption that the path loss model is the same for all the network nodes, and that shadowing/fading effects are not considered, transmitting range, and transmit power level are equivalent concepts. Since , traditionally, the function RA is defined in terms of range, instead of power. The Range Assignment problem, which was first studied in [KKKP00], is defined as follows:

Definition 2 *Let N be a set of nodes in the d-dimensional space, with $d = 1, 2, 3$. Determine a range assignment function $\overline{RA}$ such that the corresponding communication graph is strongly connected, and $c(\overline{RA}) = \sum_{i \in N} (\overline{RA})^{\alpha}$ is minimum over all connecting range assignment functions, where α is the distance-power gradient.*

The cost measure $c(RA)$ used in the definition of the RA problem is the sum of the transmit power levels used by all the nodes in the network.

Definition 3 *MINIMUM TRANSMITTING RANGE MOBILE (MTRM): Suppose n nodes are placed in $[0, l]^d$, and assume that nodes are allowed to move during a time interval $[0, T]$. What is the minimum value of r such that the resulting communication graph is connected during some fraction, f, of the interval?*

Thus, RA can be informally stated as the problem of finding a minimal nodes range assignment that generates a connected communication graph, where 'minimal' is intended as 'least energy cost'. In a certain sense, the RA problem can be seen as a generalization of the problem of determining the critical transmitting range (CTR) for connectivity.

3.2.1 Features of communication graph

It have been shown that the constraints of the critical value of the transmitting range CTR can be determined in order to guarantee the most important network property, that is, connectivity.

Definition 4 ***Connectivity*** *A graph G is said to be k-connected, where $1 <= k < n$, if for any pair of nodes u, v there exist at least k node disjoint paths connecting them. The connectivity of G, denoted as $k(G)$, is the maximum value of K such that G is K-connected. A 1-connected graph is also called simply connected.*

The interest in studying the CTR for K-connectivity is motivated by the fact that, when a network is K-connected, at most $K-1$ node or link faults can be tolerated without disconnecting the network. So, a K-connected network is more resilient to faults than a simply connected network, where a single node or link failure might partition the network.

3.2.2 Virtual topologies classification

Various research deal with services oriented architectures and communication in mobile wireless networks are interested to use virtual dynamic topologies to get a better network organization and ensure an efficient connectivity. Several studies have constructed and maintained virtual topologies in wireless networks in order to optimize data flows and exchanges. In [MH], Haddad et al proposed a classification of mostly used topology based solutions in ad hoc networks. As we can see from figure 3.1, the authors classify topologies into two main classes:

- **Covering node Sets**: This class is based on subsets of V which share some proprieties like covering property, dominating property,... It regroups Multipoints relays (MPR sets), Dominating Sets (DS), Independent Sets (IS), Minumum Independent Sets (MIS), ect
- **Link Based Structure**:This class is based on subgraphs of G. It regroups Neighborhood Graphs (NG), Spanning Tree (ST), Minumum Spanning Tree (MST), ect.

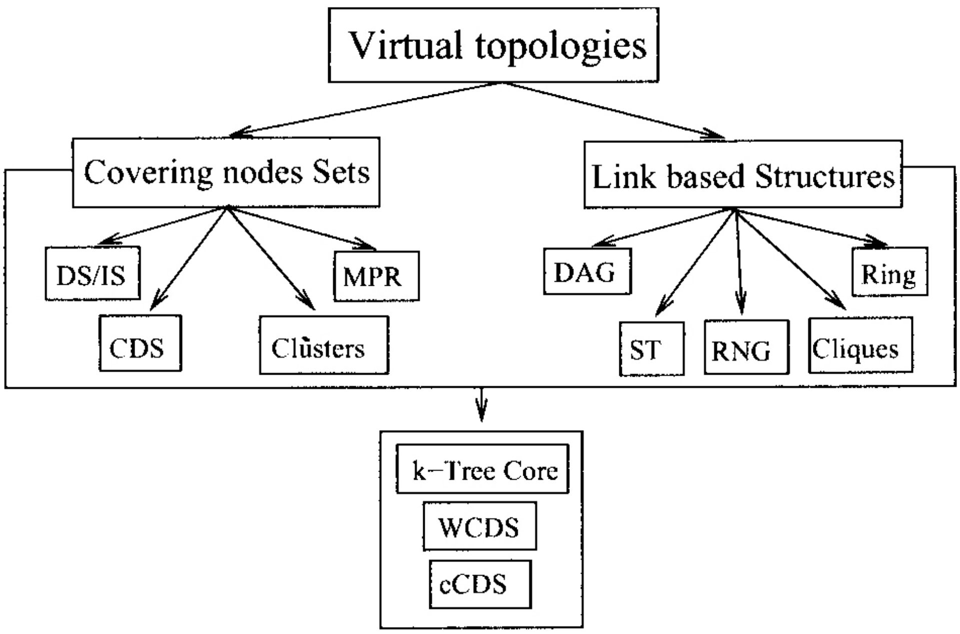


Figure 3.1: Virtual topologies classification

3.2.3 Covering node Sets

Virtual topologies of this class usually choose a subset of nodes which will be considered as network backbone. In other words, the chosen nodes will have some privileges and responsibilities according to the problem. In the case of service oriented applications, the privileged nodes should ensure the task associated to services, like: hosting services and their applications, forwarding requests, applying policies of load balancing, ect.(see Figure 3.2).

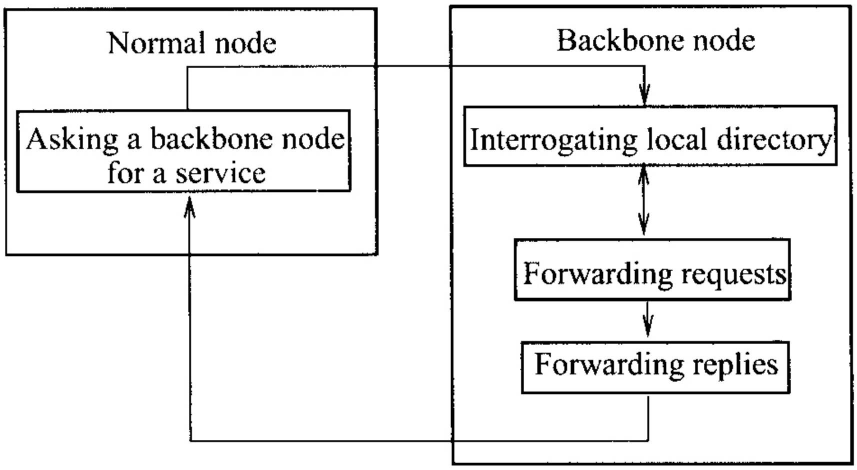


Figure 3.2: Application strategy for covering node sets.

We remark that "not backbone nodes" interrogate privileged nodes to get/access to a given service. Backbone nodes should, then, cooperate to satisfy the request. They also can construct between them a distributed directory containing all information concerning network services. In the following, we describe some topologies which are based on covering nodes:

3.2.3.1 Randomized virtual backbone construction

In many applications of WSNs, nodes alternately shut down their transceivers in order to reduce power consumption. However, a certain number of nodes must keep the radio on, in order to preserve network connectivity. Thus, active nodes must form a connected backbone. We refer to this property as 'active connectivity'. Another desirable property is that any inactive node has at least one active node within its transmitting range. In fact, inactive nodes still sense the environment, and, in case an inactive node detects an anomalous event, we want that the information regarding this event propagates quickly through the network, eventually reaching the operator. This can be accomplished only if every inactive node is able to directly communicate with at least one active node (and if the set of active nodes forms a connected backbone). Since if this property holds the set of active nodes is a dominating set,

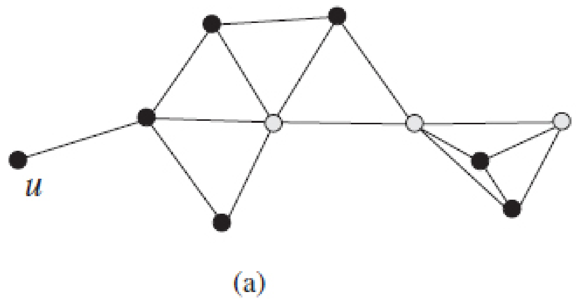


Figure 3.3: Active connectivity and active domination of the virtual backbone. Active nodes are light gray, and inactive nodes are black. The backbone of active nodes in (a) satisfies active connectivity, but not active domination (node u has no direct connection to any active node). The backbone in (b) satisfies both active connectivity and active domination.

we refer to this property as 'active domination'. Examples of virtual backbones are reported in Figure 3.3.

3.2.3.2 Dominating and Independent sets

A Dominating Set (DS) of the graph G is a subset S of V such that every node of V is either in S or adjacent to at least one node of S (i.e for each $v \in (V\S) : N(u) \cap S \neq \emptyset$). Figure 3.4 illustrates that a Minimal Dominating Set (MDS) is a dominating set such that no subset of it, satisfies the dominating property.

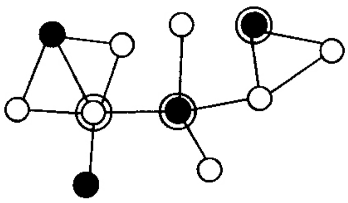


Figure 3.4: Dominating set (surrounded)/Maximal Independent Set (black)

An Independent Set (IS) in the graph G is a subset S of V such that S does not admit any pair of adjacent vertices (i.e. for each $v \in S : N(v) \cap S = \emptyset$). The independent set S is Maximal Independent Set (MIS) if there is no independent set $S\prime$ such that $S \subseteq S\prime$. For any maximal independent set, we have the following properties:

- A MIS is also a DS.

3.2.3.3 Connected Dominating sets

A Connected Dominating Set (CDS) of the graph G is a subset S of V such that:

- Nodes of S form a dominating set of G.
- The induced subgraph by the set S is connected (see figure 3.5).

Constructing a dominating set generally aims to get a better topology control strategy by reducing the network communication overhead. In [YJY06], authors proposed an energy efficient distributed connected dominating set algorithm. This algorithm prolongs the network lifetime and balances energy consumption. Some variants of CDS have been introduced by extending concept of traditional domination to d-hop domination or by seeking some fault tolerance properties (k-connection).

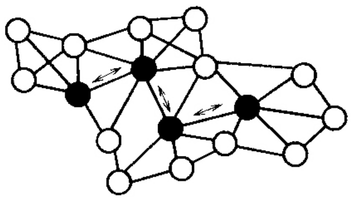


Figure 3.5: A Connected Dominating Set (CDS)

3.2.4 Link based structure

This class is based on subgraphs of the network graph. Unlike to the first one, this class is based on minimizing number of used edges. This aims to minimize the communication overhead. Indeed, the different service requests can induce a broadcast storm in the network. So, by diminishing the number of edges without losing network connectivity, the congestions due to redundant message may decrease. In the following section, we describe the most known link based topologies:

3.2.4.1 Minimum Spanning Trees

Definition 5 *(Spanning tree) Given a connected graph $G = (N, E)$, a spanning tree of G is a tree $T = (N, E_T)$ that contains all the nodes in G and is such that $E_T \subseteq E$.*

Definition 6 *(Minimum spanning tree) Given an edge-weighted graph $G = (N, E)$, a Minimum Spanning Tree (MST) for G is a spanning tree of G of minimum cost [SB02].*

Definition 7 *(Euclidean MST) Given a set N of nodes placed in the d-dimensional space $(withd = 1, 2, 3)$, and a set of edges E between these nodes, an Euclidean MST (EMST) is a MST of the edge-weighted graph $G = (N, E)$, where each edge has a weight equal to the Euclidean distance between its endpoints.*

Theorem 1 *Let N be a set of N nodes placed in $R = [0, l]^d$, with $d = 1, 2, or3$. The CTR for connectivity r_C of the network composed of nodes in N equals the length of the longest edge of the EMST T built on the same set of nodes.*

The CTR in dense networks can be characterized using results taken from a recent applied probability theory, the theory of Geometric Random Graphs (GRGs). Since the CTR equals the longest EMST edge, probabilistic solutions to the CTR problem in dense networks can be derived using results concerning the asymptotic distribution of the longest EMST edge. The construction of the range assignment by constructing an MST on the nodes as follows:

1. Let $N = u_1, ..., u_n$ be a set of points (nodes) in the two- or three-dimensional space.
2. Construct an undirected weighted complete graph $G = (N, E)$, where the weight of edge $(u_i, u_j) \in E$ is $\delta(u_i, u_j)^\alpha$.
3. Find a minimum weight spanning tree T of G.
4. Define range assignment RA_T, with $RA_T(u_i) = max_{j|(u_i,u_j)\in T}\delta(u_i, u_j)$.

Figure 3.6 shows an example of minimum spanning tree T , and the corresponding range assignment RA_T.

3.2.4.2 K-neighbors graph

Definition 8 *(K-neighbors graph) Given a set N of points in the d-dimensional space, with $d = 1, 2, 3$, and an integer k. The k-neighbors graph is the directed graph $G_k = (N, E_k)$, where $(u, v) \in E_k$ if and only if v is one of the k closest neighbors of node u.*

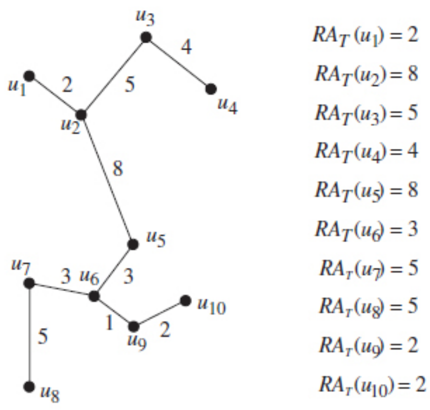


Figure 3.6: Minimum spanning tree T on the set of nodes, and corresponding range assignment RA_T.

The k-neighbors connectivity problem, is as follows: given a set N of nodes, which is the minimum value of K such that the k-neighbors graph G_k built on N is strongly connected?. Blough et al [BLRS03] proposed an approach to topology control based on the principle of maintaining the number of neighbors of every node equal to or slightly below a specific value k. The approach enforces symmetry on the resulting communication graph, thereby easing the operation of higher layer protocols.

3.2.4.3 Relative Neighborhood Graphs (RNG)

Definition 9 *(Relative neighborhood graph) Given a set N of points in the plane, the Relative Neighborhood Graph (RNG) of N is the graph $RNG = (N, E)$ such that $(u, v) \in E$ if and only if line (u, v) does not contain any other point of N in its interior, where line (u, v) denotes the moon-shaped region formed as the intersection of the two circles of radius $\delta(u, v)$ centered at u and at v (see Figure 3.7).*

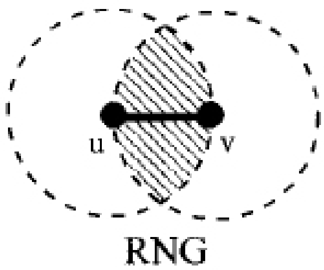


Figure 3.7: Relative Neighborhood Graph

Li et al [LWW+04] proposed a localized structure, namely, Incident MST and RNG Graph (IMRG), for topology control and broadcasting in wireless ad hoc networks. In the construction algorithm, each node first builds a modified relative neighborhood graph (RNG'), and then informs its one-hop neighbors of its incident edges in RNG'. Each node then collects all its one-hop neighbors and the two-hop neighbors who have RNG edges to some of its one-hop neighbors, and builds an Euclidean minimum spanning tree of these nodes. Each node u keeps an edge uv only if uv is in the constructed minimum spanning tree.

3.2.5 Mixed structures

A skillful virtual topology is the one which combines the advantages of the two previous classes. In other words, a mixed topology chooses a best set of nodes which dominate the remaining nodes with a minimum number of links. Indeed, a good virtual topology should offer: well connectivity , small diameter (to minimize virtual distance between each pair of nodes), a good regularity, weak energy consumption, and a fair behavior towards wireless medium access, we can find several mixed topologies such as k-tree core, weakly connected dominating sets, cliques, ect. In addition to these characteristics, it is necessary to construct and maintain the topology in reasonable message and time complexity. Interested reader can refer to [Jav08], [MH].

3.3 Topology Control Protocols

Topology Control (TC) is a well known technique used in wireless ad hoc and sensor networks to reduce energy consumption. This technique coordinates the decisions of network nodes about their transmission power to save energy, prolong network lifetime, and mitigate MAC-level medium contention, while maintaining network connectivity. For this reason, energy conserving protocols at the MAC, routing, and upper layers have been proposed [[CT00], [JV02], [KB05]]. Further energy can be saved if the network topology it-self is energy-efficient, i.e., if the nodes' transmitting ranges are set in such a way that a target property (e.g., connectivity) of the resulting network topology is guaranteed, while the global energy consumption is reduced. A protocol that attempts to achieve this is called a topology control protocol. Several examples of topology control mechanisms have been proposed [[BHM06], [BJ+02], [HSSJ02], [RM99a], [WLBW01]].

Santi [San05] proposed a coherent taxonomy for the diverse topology control protocols. The classification is depicted in Figure 3.8. Nonhomogeneous topology control is classified into three categories, depending on the type of information that is used to compute the topology; location-based approaches, direction-based approaches, and neighbor-based. In location-based approaches, it is assumed that the most accurate information about node positions (the exact node location) is known. This information is either used by a centralized authority to compute a set of transmitting range assignments that optimizes a certain measure, or it is exchanged between nodes and used to compute an optimal topology in a fully distributed manner. Typically, location-based approaches assume that network nodes, or at least a significant fraction of them, are equipped with GPS receivers. In direction-based approaches, it is assumed that nodes do not know their position but they can estimate the relative direction of their neighbors. In neighbor-based techniques, nodes are assumed to have access to a minimal amount of information regarding their neighbors, such as their ID, and to be able to order them according to some criterion (e.g., distance, or link quality). Neighbor-based techniques are probably the most suitable for application in mobile ad hoc networks. In this section, we give short description of some approaches:

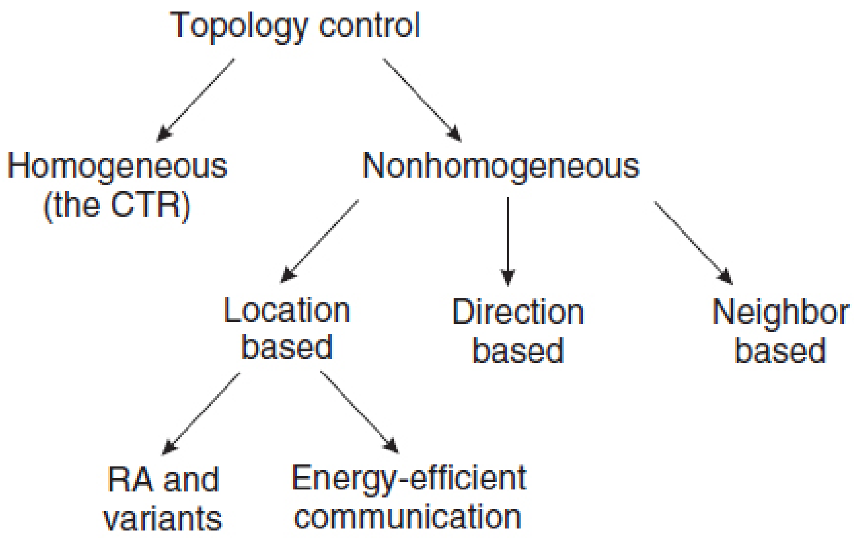


Figure 3.8: A taxonomy of topology control techniques

The KNeigh protocol The KNeigh protocol proposed in [BLRS03] is neighbor-based approache. Initially, every node broadcasts its ID at maximum power (we assume that all the nodes have the same maximum transmit power P_{max}, and that the wireless medium is symmetric). Upon receiving broadcast messages from other nodes, every node keeps track of its neighbors, storing for each of them the estimated distance. After all the initial messages have been sent, every node in the network knows its neighbor set and the distance-based ordering of the neighbors. Given this information, every node computes its k-closest neighbors list KN, and broadcasts this information at maximum power. By exchanging neighbor lists, nodes are able to determine the set of symmetric neighbors and to exclude the asymmetric neighbors from KN. At the end of the protocol execution, $KN(u)$ contains the list of neighbors of node u in the final topology, and the broadcast transmit power of node u is set to the minimum value needed to reach the farthest node in $KN(u)$. Note that this value can be computed given the received signal strength of the messages sent by the farthest node in $KN(u)$.

XTC Protocol XTC Protocol presented in [WZ04] is a neighbor-based topology control algorithm. XTC can be considered as a generalization of KNeigh: similar to KNeigh, nodes first establish an order on their neighbor nodes; then, they exchange information about the neighbor orders; finally, they compute their local view of the final network topology. The main differences between XTC and KNeigh are the following:

1. The neighbor order is based on the concept of link quality, rather than distance as it was the case in KNeigh.
2. When exchanging neighbor lists and computing the final topology, nodes consider the entire neighbor set, and not the first k elements in the order as was the case in KNeigh.

R&M protocol The R&M protocol [RM99b] is a location-based approach that builds an optimal topology for the all-to-one communication pattern, where one of the network nodes is designated as the master node, and all the other nodes send messages to the master. This traffic pattern is typical of WSNs, where the deployed sensors must send the collected data to one (or more) base station(s).

CBTC Protocol The CBTC (Cone-based Topology Control) protocol [LHB+01] is based on on the ability of the nodes to estimate the relative direction of their neighbors. CBTC sets the transmit power level of node u to the minimum value $p_{u,p}$ such that u can reach at least one node in every cone of width ρ centered at u. In other words, a node must retain connections to at least one neighbor in every direction, where parameter ρ determines the granularity of what is meant by every direction (see figure 3.9).

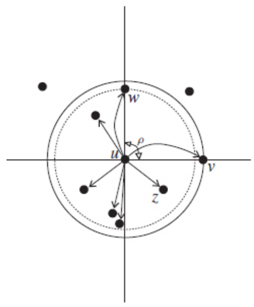


Figure 3.9: Intuition behind the CBTC protocol.

3.4 A novel social perspective of topology optimization problem

An important aspect to be considered in the design of topology control protocols is the type of information used by the nodes to build the local view of the topology: nodes can use high-quality information (e.g. neighbor node locations), medium-quality information (e.g.directional information, or distance to neighbors), or low-quality information (number and identity of neighbor nodes). In general, there is a direct relationship between information quality and energy efficiency of the computed topology: the more accurate the information available to the nodes, the more energy savings can be achieved. However, information quality must be carefully traded off with the cost incurred for making the information available to the nodes. The cost is due to either some additional hardware required on the nodes (e.g. low-power GPS receivers in case of location information) or the message overhead needed to produce/update high-quality information, or both. Therefore, in our proposal, we aim to extract realistic patterns of human mobility including repeating daily activities and social behaviours in order to provide helpful information in designing efficient topologies for communication and developing a wide range of applications enabled by mobile networking, such as routing in Mobile Ad Hoc Networks (MANETs). Further, leveraging human mobility feature to optimize realistic applications for mobile wireless networks is vital in computational social science and mobile wireless networks [NDST14]. In our view, social layer consists of one single layer that is added onto the top of the MANET layer stack positioned upper the topological layer and physical layer (see figure 3.10) [DB09].

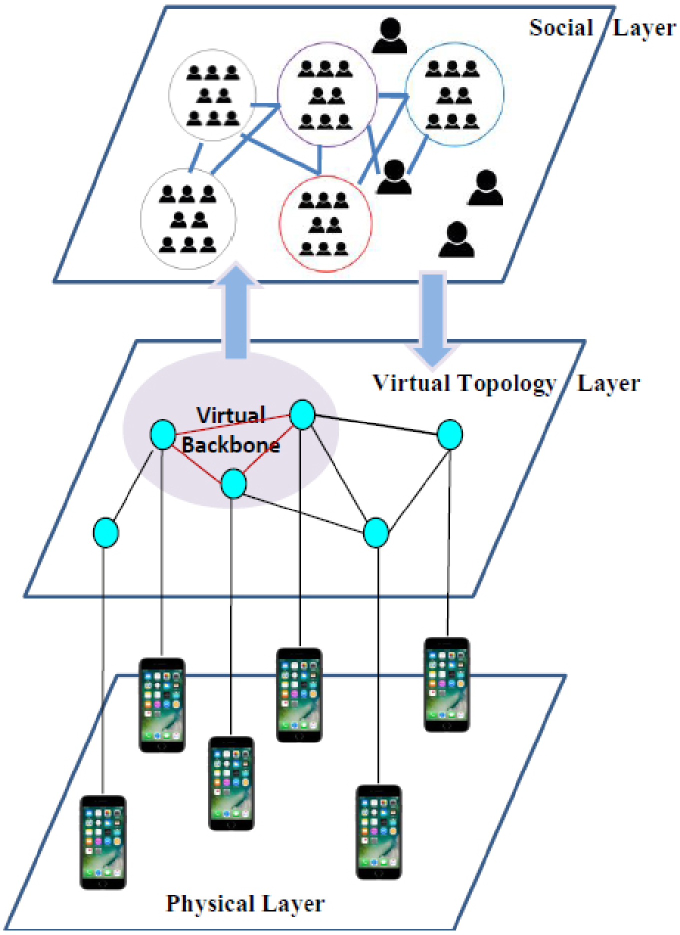


Figure 3.10: Our view of Social Layer in the protocol stack.

3.4.1 Virtual topology layer and routing

The routing layer is responsible for finding and maintaining the routes between source/destination pairs in the network: when node u has to send a message to node v, it invokes the routing protocol, which checks whether a (possibly multihop) route to v is known; if not, it starts a route discovery phase, whose purpose is to identify a route to v; if no route to v is found, the communication is delayed or aborted. The routing layer is also responsible for forwarding packets toward the destination at the intermediate nodes on the route. The topology layer, which creates and maintains the list of the immediate neighbors of a node, can trigger a route update phase in case it detects that the neighbor list is considerably changed. In fact, the many leave/join in the neighbor list are likely to indicate that many routes to faraway nodes are also changed. So, instead of passively waiting for the routing protocol to update each route separately, a route update phase can be triggered, leading to a faster response time to topology changes and to a reduced packet-loss rate. On the other hand, the routing layer can trigger the reexecution of the topology control protocol in case it detects many route breakages in the network, since this fact is probably indicative that the actual network topology has changed a lot since the last execution of topology control [San05].

3.4.2 Social layer and virtual topology layer

Human behavior is often complex and context-dependent. The social layer lies in the assumption that human mobility can be explained by several mobility patterns that depend on a sub-set of the contextual variables and these can be learned by a simple models. Therefore, it is very interesting to model approaches that takes as input certain spatial, temporal and social parameters to provide valuable scheme to the topological and physical layers. In this social perspective, we aim to analyze the influence of human habits on the movement of people and detect the individuals with strong similar interest who move towards the same geographical area, in other word, we aim to extract a relevant knowledge from the pattern correlation of individuals (community structure). Therefore, understanding the network community structure is a powerful means of representing patterns of connections or interactions between individuals. It not only provides helpful information in developing more social-aware strategies for social network problems but also promises a wide range of applications enabled by mobile networking, such as routings in Mobile Ad Hoc Networks (MANETs) and worm containments in cellular networks.
This finding allows to topological layer to construct an efficient and reliable topology which is not only dependent on the relative locations and physical connections of mobile entities but also on the individual's behaviour and his/her daily communities.

3.4.3 Physical layer and virtual topology layer

The MAC (Medium Access Control) layer is responsible for regulating the access to the wireless, shared channel. Medium access control is of fundamental importance in ad hoc/sensor networks in order to reduce conflicts as much as possible, thus maintaining the network capacity to a reasonable level. Thus, the interaction between the MAC layer and topology control is as follow: when node u wants to send a packet to node v, it first sends a Request To Send control message (RTS), containing its ID, the ID of node v, and the size of the data packet. If v is within u's range and no contention occurs, it receives the RTS message, and, in case communication is possible, it replies with a Clear To Send (CTS) message. Upon correctly receiving the CTS message, node u starts the transmission of the DATA packet, and waits for the ACK message sent by v to acknowledge the correct reception of the data. In order to limit collisions, every node maintains a Network Allocation Vector (NAV), which keeps trace of the ongoing transmissions. The NAV is updated each time a RTS, CTS, or ACK message is received by the node. Note that any node within u's and/or v's transmitting range overhears

at least part of the RTS/CTS/DATA/ACK message exchange, thus obtaining at least partial information on the ongoing transmission. Using different transmit power levels can introduce additional opportunities for interference between nodes. On the other hand, using reduced transmit powers can also avoid interference. To clarify this point, consider the situation depicted in Figure 3.11, there are four nodes u, v, w, and z, with $\delta(u,v) = d_1 < d_2 = \delta(v,w)$ and $\delta(w,z) = d_3 < d_2$. Node u wants to send a packet to v, and node w wants to send a packet to z. Assume all the nodes have the same transmit power, corresponding to transmitting range r, with $r > d_2 + max\{d_1, d_3\}$. Then, the first between nodes v and z that sends the CTS message inhibits the other pair's transmission. In fact, nodes v and z are in each other's radio range, and overhearing a CTS from v (respectively, z) inhibits node z (respectively, v) from sending its own CTS. Thus, with this setting of the transmitting ranges, no collision occurs, but the two transmissions cannot be scheduled simultaneously.
Assume now that nodes u and v have radio range equal to r_1, with $r_1 = d_1 + \epsilon < d_2$ and that nodes w and z have range r_2, with $r_2 > d_2$. In this situation, w and z cannot hear the RTS/CTS exchange between nodes u and v and they do not delay their data session. However, when node w transmits its packets, it causes interference at node v, which is within w's range. Thus, in this case, using different transmit powers creates an opportunity for interference.

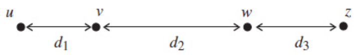


Figure 3.11: The importance of appropriately setting the transmit power levels.

Finally, assume nodes u and v have radio range r_1, and nodes w and z have range equal to r_3, with $r_3 = d_3 + \epsilon < d_2$ With these settings of the radio ranges, the two transmissions can occur simultaneously, since node v is outside w's radio range and node z is outside u's radio range. In this case, using different power levels reduces the opportunities for interference, leading to an increased network capacity.

Santi [San05] explained that the task of setting the transmit power levels should be performed by the topology control layer, which, having a network wide perspective, can take the correct decisions about the node's transmitting range. On the other hand, the MAC layer can trigger reexecution of the topology control protocol in case it discovers new neighbor nodes. The MAC level can detect new neighbors by overhearing the network traffic and analyzing the message headers; this is by far the fastest way to discover new neighbors, and a proper interaction between MAC and topology control ensures a quick response to changes in the network topology.

3.5 Conclusion

By exploiting ad hoc wireless technology, various portable devices (cellular phones, PDAs, laptops, pagers, and so on) and fixed equipment (base stations, wireless Internet access points, etc.) can be connected together, forming a sort of ubiquitous network. Deploying virtual topology in mobility studies, helps to investigate the dynamic behavior of the network and leads us to develop new approaches for dealing with current revolution in Internet communications.

CHAPTER 4

Community Detection Methods in Complex Networks

4.1 Introduction

In recent times, the computer revolution has provided scholars with a huge amount of data and computational resources to process and analyze these data. The size of real networks one can potentially handle has also grown considerably, reaching millions or even billions of vertices. The need to deal with such a large number of units has produced a deep change in the way graphs are approached. The community detection methods has brought significant advances to our understanding of complex systems. Further, the community detection methods focus on the partition quality of such graphs in graphs [GN02], [NG04] instead of a simple cut size of graph or clustering approach [New10].

In complex networks, the communities are groups of nodes which share probably a common proprieties and/or similar functions. The communities may be correspond , for example, to groups of Web pages accessible over the Internet that have the same subject [FLGC02], functional modules as cycles and pathways in metabolic networks [GA05],[PDFV05], a set of people or groups of people with sme pattern of contacts or interactions between them [GN02],[LN04], subdivisions in the food webs [Pim79], [KFM+03] and so on. The main objective of community detection methods is to discover a relevant community structure, for example figure 4.1 illustrates the network structure of the Web site of Ferhat Abbas University (Algeria). As we can see from the figure, the network contains six communities identifying groups of users with similar behaviour which can help us to personalize the Web site and improve its design.

The community discovery methods have attracted considerable interest and curiosity from

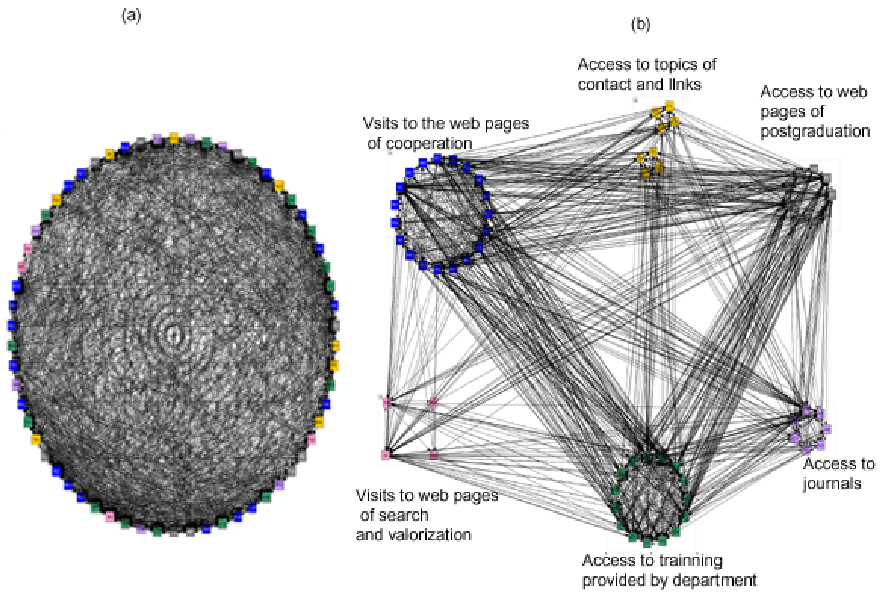


Figure 4.1: a) Modeling the network structure of the Web site of Ferhat Abbas University (Algeria). The nodes represent resources and edges represent the browsing sequences of users during each session [SMLD11]. b) We have obtained six communities extracted from the users' behavior. communities (labeled by colors) were detected by applying Fast algorithm[New04].

the science community in recent decades. It would clearly be beneficial if discovering communities can be used in same manner to characterize and analyze the behavior of users in Mobile Wireless Networks.

In this chapter, we point out the most cited works in order to provide a taxonomy for community detection methods. As the research on community identification has grown quite popular and the number of published proposals for community discovery algorithms as well as reported applications is high, we do not even pretend to be able to give an exhaustive survey of all the existing methods, but rather an explanation of the methods commonly applied. We give definitions of community structure in section 4.2. In section 4.3, we propose a taxonomy of the detection community methods. Section 4.4 describes agglomerative methods. Section 4.5 survey divisive methods. In section 4.6, we explain the key idea of the dynamic community detection methods.

4.2 Communities description

Despite the large amount of study in this area, a consensus on what is the definition of community has not been reached. Conceptually, the definitions can be separated into two main categories, self-referring and comparative definitions. The concept of subgraph is the central to all such definitions.

4.2.1 Graph concepts

We consider a graph $G = (V, E)$, with V is the set of vertices and E is the set of edges. Vertices are also known as nodes, points and as actors, agents or players in social networks. $A_{i,j}$ is the adjacency matrix of the network . It is 1 if nodes u and v are connected, otherwise it is 0, and k_i is the degree of node i. In an undirected graph,each edge is an unordered pair v, w. In a directed graph, edges are ordered pairs. We denote the number of nodes in a graph n, and denote the number of edges m. Graph is used to represent all complex networks.

4.2.2 Comparative definitions

The comparative definitions consist in comparing the internal links with external links. Authors have proposed a different similarities measures between nodes to discover the community structure. The intuitive notion of similarities derives from the relative strength, frequency, density, or closeness of links within a subgroup, and the relative weakness, infrequency, spareness, or distance of links to discover subgroup members and non members. A community may be defined as LS-set. An LS-set is a set of nodes such that each of its proper subsets has more ties to its complement within the set than outside [WF94].

Raddichi et al. [RCC+04] have defined a community in the weak sense as a set of nodes whose total number of internal links is greater than the total number of links to the outside. Thus, a subgraph C is a community in a strong sense if

$$k_i^{in}(C) > k_i^{out}(C) \forall i \in C \tag{4.1}$$

The subgraph C is a community in a weak sense if

$$\sum_{i \in C} k_i^{in}(C) > \sum_{i \in C} k_i^{out}(C) \tag{4.2}$$

where $k_i^{in}(C) = \sum_{j \in C} A_{i,j}$ is the number of edges connecting node i to other nodes belonging to C, and $k_i^{out}(C) = \sum_{j \notin C} A_{i,j}$ is the number of connections toward nodes in the rest of the network.

4.2.3 Self-referring definitions

The self-referring definitions consider the subgraph alone. It identifies classes of subgraphs like cliques, n-cliques, k-plexes, etc... They are maximal subgraphs, which cannot be enlarged with the addition of new vertices and edges without losing the property which defines them. Self-referring definitions can include also the definition introduced in[New06] that defines a community as indivisible subgraph.

4.2.4 Quality Functions

How can we know if detected communities are good or no and how to value such partitions? What is the better partition for the network under study? To answer to such questions, Newman and Girvan [NG04] have introduced a measure of quality of partition which they called modularity. The modularity is based on assortative mixing measure proposed by Newman [New03]. The modularity is defined as:

Definition 10 *Suppose a particular division of network to k communities, this can be represented by a $k \times k$ symmetric matrix e which each element e_{ij} is the fraction of all network edges that link vertices in group i to group j.*

Trace matrix $Tr(e)$ represents a fraction of network edges that connect the vertices in a group and obviously a good division has a high value of $Tr(e)$.

$$Tr(e) = \sum_i e_{ii} \tag{4.3}$$

Although this value alone is not a good measure of the quality, because placing all vertices in a single group would give the maximal value 1 whereas no information of community structure is provided.

a_i^2 is the expected fraction of edges within community i when the edges were distributed randomly on the network. In addition, the real fraction of links exclusively within a partition is e_{ii}. Thus, we can compare the two quantities and sum over all the partitions in the graph.

$$Q = \sum_i (e_{ii} - a_i^2) = Tre - ||e^2|| \tag{4.4}$$

Where: $||e||$ is the sum of matrix e elements. This quantity measures the fraction of the within-community edges in the network minus the expected value of the same quantity in a network with the same community divisions but when connections between nodes are random. Values different than 0 indicate deviations from randomness, and values above 0.3 indicate a good modular structure [New04]. In practice, values above 0.7 are uncommon, and indicate a very clear structure. However, we may find that high modularity values doesn't indicate a significant community partitions, and it is up to the network under study [FB07].

In order to understand the resolution limit of the quality functions in graph, we have conducted our experiments on several graphs of different size [DBS13] [SMLD12a] [SMLD12b] . Here, we give two examples; first, we have extracted the activities of the website of Setif university, after that, we have visualized the graph that shows the original network structure (it contains 63 nodes and 1214 edges), and we have also used a benchmark of 1854 nodes. The results of modularity on both graphs are given in table 4.1, and table 4.2. Thus, we remark that some methods find the desired communities but with very small modularity values. As matter of fact, several works have been proposed based on local modularity maximization in order to find community structure more correctly either for small graphs or large one.

Table 4.1: Results of applying community detection methods on small graph

Community Method	Communities number	Quality Function
Newman, 2006	04	0.138
Reichardt et al, 2004	04	0.141
Pons et Latapy, 2006	08	0.116

Furthermore, a high value of modularity does not necessarily mean that the graph has community structure [For10]. According to the definition of modularity, random graphs are supposed to have no community structure, as the linking probability between vertices is either constant or a function of the vertex degrees, so there is no bias a priori towards special groups of vertices. Still, random graphs may have partitions with large modularity values.

Table 4.2: Results of applying community detection methods conducted on banchmark of 1854 nodes

Community Method	Communities number	Quality Function
Newman, 2006	13	0.572
Reichardt et al, 2004	15	0.624
Pons et Latapy, 2006	19	0.721

4.3 Taxonomy of community discovery methods

The problem of community detection get close to traditional clustering approaches and partitioning graph methods in computer science. The typical problem in computer science is that of dividing the vertices of a network into some number of groups with roughly equal size, while minimizing the number of edges that run between vertices in different groups. Graph partitioning problem arises for example in the optimal allocation of processes to processors in a parallel computer. In practice, most approaches to graph partitioning have been based on iterative bisection. Authors [KL70] [PAP+90], [Fie73] found the best division of the complete graph into two groups, and then further subdivided those two groups until they have the required number of groups. Division into more than two groups can be achieved by repeated bisection, but there is no guarantee that the best division into three groups can be arrived at by finding the best division into two and then dividing one of those two again. Furthermore, these algorithms give no hint about when we should stop the repeated bisection process, that is, about how many communities there should be in a network [New04]. For these reasons, the traditional graph partitioning methods are not ideal for analyzing general network data. In addition, it has been shown that data clustering methods are not best suited to the processing of the large data, due to the fact that, several clustering methods are not able to deal with properties of complex networks. Therefore, several community detection approaches have been proposed in order to identify a significant community structure in complex networks.

In our work [DB14], we have reviewed the community detection methods and we have suggested a classification of theses methods(see Fig.2 and Fig.3). The proposed taxonomy help us to make conclusions about how to investigate the useful information incorporating in the complex network under study to understand its community structure. Our taxonomy is based onthe three following points of view:

A. Agglomerative versus divisive

According steps employed when identifying nodes within groups,

Jain and Dubes [JR88] have defined two distinct approaches of clustering methods: agglomerative and divisive, they are based on the manner employed to identify nodes within groups. Community discovery methods use also either agglomerative or divisive approaches:

- Bottom-up or agglomerative algorithms start with each node in its own singleton community or another set of small initial communities, and they merge iteratively these communities into larger ones.

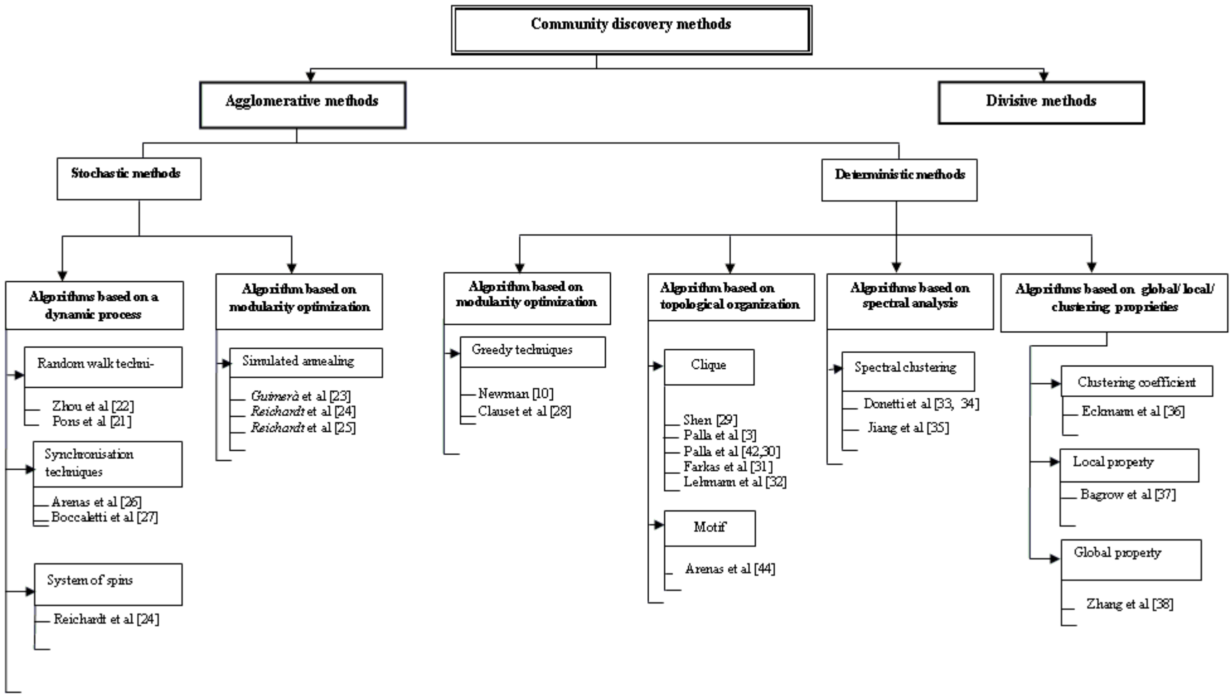


Figure 4.2: Community discovery methods: a. Agglomerative methods.

- Top-down or divisive algorithms split iteratively or recursively the network into smaller and smaller communities.

B.Stochastic versus deterministic

The deterministic model has no stochastic elements and the entire input and output relation of the model is conclusively determined. A stochastic model has one or more stochastic elements and it give efficient results, but there are several cases for which it is difficult to build an intuitive perspective for the stochastic system.

C. Various computer implementations of community discovery methods

We classify the community detection methods with respect to the process used to model links and nodes such as random process, synchronization, etc..., and we consider how does the approach characterize the properties of the network to detect communities(clustering coefficient, betweeness,..., etc).

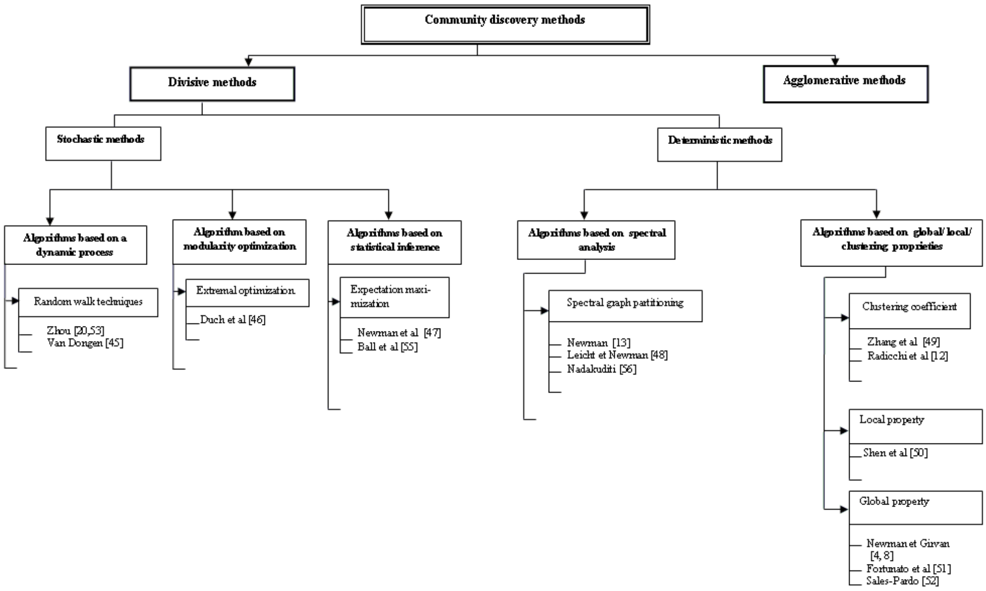


Figure 4.3: Community discovery methods: a. Separative methods.

4.4 Agglomerative methods

A wide range of agglomerative hierarchical community detection algorithms have been proposed. They begin with nodes in n different communities and group together communities that are the most similar. In this section, we will describe some agglomerative approaches, we give a classification with respect to the process employed during the discovery steps. Figure . 4.4 shows the community structure of an agglomerative method; label propagation method [RAK07]. We have applied it on the southern women network [DGG09]. As we can see from the figure, this network contains 18 women who attended 14 different events, and the result show three communities and their memberships who have a similar behavior.

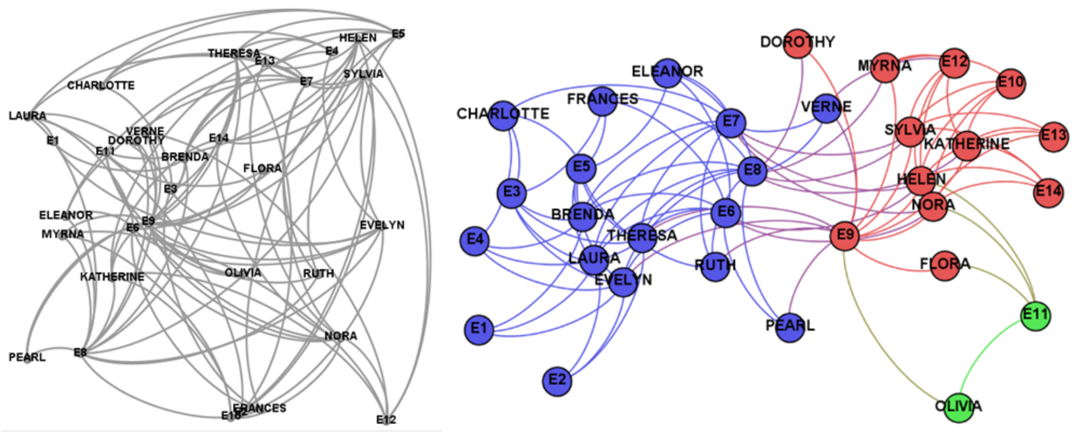


Figure 4.4: a)The Southern women dataset [DGG09] b) Partition of the bipartite graph into communities aplying label propagation algorithm. The nodes labeled E represent the events the women attended.

4.4.1 Algorithms based on a dynamic process

In this section, we will describe: spin-spin interactions, random walk techniques and synchronization techniques.

4.4.1.1 Random walk techniques

Zhou and Lipowsky [ZH04] have proposed a random walk method based on the average number of steps a Brownian particle (random movement of a particle) takes to reach a given node from a source node. The authors have used the Brownian movement to develop the Netwalk algorithm (NW) based on the proximity index. The proximity index for each nearest-neighboring pair of nodes i and j is defined as:

$$\Lambda(i,j) = \frac{\sqrt{\sum_{K \neq i,j}^{N} [d_{ik} - d_{jk}]^2}}{(N-2)} \tag{4.5}$$

If two nearest-neighboring nodes i and j belong to the same community, then the mean-first-passage-time $d_{i,k}$ from i to any another node k will be approximately equal to that from j to k. So, it will be small if i and j belong to the same community and large if they belong to different communities. Initially, the Netwalk algorithm considers each node as a single community. Then it merges the two communities with the lowest proximity index into a single community and then updates the proximity index between this new community and all the other remaining communities that are connected to it. This merging process is continued until all the nodes are merged into a single community corresponding to the whole network. The time complexity is $\mathcal{O}(n^3)$, thus it will be inadmissible to apply this method on large graphs.

Pons et al. [PP06] have proposed Walktrap algorithm based on random moves, in other words, when a walker is within a community it has a strong probability of remaining in the same community at the next step. The authors have defined distance measurement which is related to the spectral approaches, as a matter of fact, two close nodes belonging to the same community will have similar components on the principal eigenvectors. The algorithm computes the connected components, and applies an agglomerative algorithm to discover communities on connected sub-graphs. This algorithm gives better performance in $\mathcal{O}(nmlog(n))$, however, if the length of the steps are big, the quality of the results will decrease.

4.4.1.2 Synchronization techniques

Several studies have proposed a synchronization process to discover communities based on the idea of relationship between topological scales and dynamic time scales in complex networks [ADGPV06], [BIL+07]. It have been shown that high densely interconnected sets of oscillators, such as oscillators are placed on nodes of graph, synchronize more easily that those with sparse connections [ADGPV06]. The scenario starts from random initial conditions such as the highly interconnected oscillators will synchronize first to form local clusters, whereas a full synchronization requires a longer time. This process occurs at different time scales when a significant community structure exists. In [BIL+07], the time complexity of the algorithm is $\mathcal{O}(mn)$, or (n^2) on sparse graphs, and gives good results on ground truth graphs. However, synchronization-based algorithms may not be reliable when the size of communities is different.

4.4.1.3 System of spins

Reichardt and Bornholdt have shown that it is possible to reformulate the problem of community detection as the problem of finding the ground state of a spin glass model [RB04]. They have proposed a detection community algorithm based on Potts models with Q states. Potts model is one of the most common models used in statistical physics in order to describe the behavior of the magnetic bodies [KF72], thus bodies are modeled as spins of Q states located at nodes of the network, and there are interactions between neighbors. The basic principle of the model is that edges should connect vertices of the same class (i. e. same spin state), whereas vertices of different classes (i. e. different spin states) should be disconnected.
Fu and Anderson [FA86] showed, by analogy with ferromagnetic system, that there is a relation between the energy of the physical systems , which is represented by the Hamiltonian, and the cost function in optimization combinatorial problem. Thus,the Hamiltonian of a spin is written as:

$$H = -J \sum_{(i,j) \in E} \delta_{\sigma_i \sigma_j} + \gamma \sum_{s=1}^{q} \frac{n_s(n_s - 1)}{2} \tag{4.6}$$

Where: E is the set of edges, $\sigma_i (i = 1.., n)$ denotes the individual spins which can take q values, n_s indicates the number of spins that have s spin such as $\sum_{s=1}^{q} n_s = N$, j is a constant expressing the coupling strength, δ: is a positive parameter, and, σ is the Kronecker delta. Each node is characterized by a spin which can have q possible values. The first sum is the standard ferromagnetic Potts term which represents a homogeneous distribution of the spins in the network, and it can be written $H_{ferr} = -JM$. The second term sums up all the possible

pairs of spins which have equal values. It represents the diversity of the configuration of spins or the existing classes of spins.
Therefore, the communities correspond to the classes of nodes having equal values of spin. The q number of possible spins corresponds to the maximum number of communities which we can detected. The authors use Monte Carlo algorithm to detect the community structure. The Hamiltonian is computed using simulated annealing algorithm[SKP83]. The algorithm is able to detect the overlapped communities and allows the quantification of the communities' stability.

4.4.2 Algorithms based on modularity optimization

It have been shown that high values of modularity indicates good partitions [New03]. The best partition , or at least a very good one, fits with the maximum value of modularity. This is the main motivation for having many modularity maximization based methods.

4.4.2.1 Simulated annealing

Guimer et al. [GSPA04] have shown that finding the modularity of network is analogous to finding the ground-state energy of a spin system. They have used a simulated annealing algorithm for modularity optimization. The optimization is based on local moves and global moves. In fact, a single node is shifted from one cluster to another, which is taken randomly, and the global moves consist of merging and splitting of communities. Simulated annealing algorithm converges generally more closely towards the optimal solution but it is more efficient for small networks.

4.4.2.2 Greedy techniques

In the work [New04], Fast algorithm is based on a greedy optimization. It starts with each node being the single member of each community, it repeatedly joins together the two communities whose amalgamation produces the largest increase in Q but don't join the pair of communities which don't have links between them. For a network of n nodes, after $(n-1)$ such joins, the algorithm stops when the results of the merging process is a single community. Thus, the total running time is $\mathcal{O}(mn)$, or $\mathcal{O}(n)^2$ on a sparse graph. Clauset et al [CNM04] have improved fast algorithm, therefore, it performs well in $\mathcal{O}(nlog^2 n)$, and computes changes in modularity in order to find a pair of communities i, j with the largest ΔQ_{ij}. Fast algorithm is based on local information to identify communities, however, the community structure isn't a local quantity.

4.4.3 Algorithms based on topological organization

Several detection communities approaches are based on the observation that community can be interpreted as a smaller complete (fully connected) subgraphs that share nodes. In this section, we describe the most cited methods that belong to the topological organization class.

4.4.3.1 Clique

Palla et al [PDFV05], [DPV05] have defined a method based on percolation process to detect the overlapped communities in the networks. The clique percolation method (CPM) uses the high density of the internal edges that form cliques. Two k-cliques are adjacent if they share $(k-1)$ nodes, and a community is defined as a union of all the K-cliques that can be reached using chains of adjacent k-cliques. Such communities can be better visualized using a k-clique template which is an isomorphic object for a complete graph of k-nodes. This object can be placed on a k-clique of the network and be moved towards the adjacent k-clique by changing one of its nodes and keeping its others $(k-1)$ nodes. Thus, the community (k-clique percolation cluster) is the largest connected subgraph which can be entirely explored while rolling the object on the k-clique. An extension of CPM algorithm have been proposed for the weighted networks [FAPV07] and the directed networks [PFP+07]. First of all, the algorithm computes the maximal cliques from k-cliques to detect communities. Detecting maximal cliques is known to require a running time that grows exponentially with the size of the graph. However, the authors found that, for the real networks they analyzed, the procedure is quite fast, due to the fairly limited number of cliques, and that sparse graphs with up to 10^5 vertices can be analyzed in a reasonably short time. The actual scalability of the algorithm depends on many factors, and cannot be expressed in closed form.

Lehmann et al [LSH08] have addressed the problem of community detection on bipartite networks. They proposed a biclique community detection algorithm. In this case one uses bipartite cliques, or bicliques : a subgraph $K_{a,b}$ is a biclique if each of a vertices of one class are connected with each of b vertices of the other class [DPV05]. However, finding all the bicliques of a graph is an NP-complete problem, mostly because the number of bicliques tends to grow exponentially with the size of the graph.

In [SCCH09], Shen et al. proposed an algorithm to detect both the overlapping and hierarchical properties of complex community structure together. This algorithm deals with the set of maximal cliques and adopts an agglomerative framework. The quality function of modularity is extended to evaluate the goodness of a cover. Initially, the algorithm finds all the maximal cliques using Bron-Kerbosch algorithm [BK73]. Then each maximal cliques and each subordinate nodes are determined as an initial communities. Then, the algorithm repeatedly computes the similarity between each pair of communities and join together the two communities whose amalgamation produces a maximum similarity. The similarity between two communities C_1 and C_2 is defined as:

$$M = \frac{1}{2m} \sum_{v \in C_1, w \in C_2, v \neq w} [A_{vw} - \frac{k_v k_w}{2m}] \tag{4.7}$$

Where: A_{vw} is the adjacency matrix, k_v (k_w resp) is the degree of the node v (w resp), and $m = \frac{1}{2} \sum_{vw} A_{vw}$ is the total number of the edges in the network. The extended modularity is written as:

$$EQ = \frac{1}{2m} \sum_i \sum_{v \in C_i, w \in C_i} \frac{1}{O_v O_w} [A_{vw} - \frac{k_v k_w}{2m}] \tag{4.8}$$

Where: O_v is the number of communities including node v. The algorithm determines the suitable cut on the dendrogram with the maximum value of EQ. EAGLE algorithm don't have only the capacity to detect sub communities until none of them can be divided (indivisible graph), but also it detects the overlapped communities. However, the algorithm is time-consuming because it finds out all the maximal cliques in the complex network.

4.4.3.2 Motif

The high density of edges within a community determines correlations between nodes going beyond nearest-neighbours, and which are indicated by the presence of motifs. Arenas et al [AFFG08] have shown how motifs can be used to define general classes of nodes, including communities, by extending the mathematical expression of Newman-Girvan modularity. They have defined the motif modularity as the fraction of motifs inside the communities minus the fraction in a random network which preserves the nodes' strengths.

4.4.4 Algorithms based on spectral analysis

Donetti et al. [DM04] proposed an approach based on the spectral properties of the Laplacien matrix. since the values of the eigenvector components are close for vertices in the same community, one can use them as coordinates, such that vertices turn into points in a metric space. So, if one uses M eigenvectors, one can embed the vertices in an M-dimensional space.

Communities appear as groups of points well separated from each other, as illustrated in Figure 4.5. The separation is the more visible, the larger the number of dimensions/eigenvectors M.

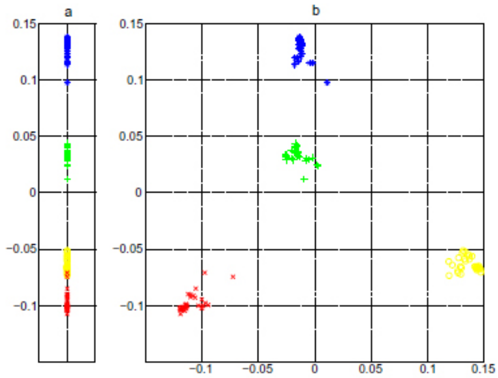


Figure 4.5: Example of a division of graph in four communities, indicated by the colours. The communities are better separated in two dimenions (b) than in one (a).

An improvement approach have been proposed using a normalized Laplacien matrix [DM05]. In [JDY09], Jiang et al have rewritten the modularity using the spectral clustering in order to maximize the modularity and, thus, correctly detect the community structure in the network. The spectral methods give a good results but the eigenvectors computing is time-consuming.

4.4.5 Algorithm based on clustering properties

Eckmann et al.[EM02b] have proposed a method based on the clustering properties. The idea is to use the clustering coefficient of a vertex as a quantity to distinguish tightly connected groups of vertices. Many edges mean many loops inside

a community, so the vertices of a community are likely to have a large clustering coefficient. The latter can be related to the average distance between pairs of neighbours of the vertex. The possible values of the distance are 1 (if neighbors are connected) or 2 (if they are not), so the average distance lies between 1 and 2. The more triangles there are in the subgraph, the shorter the average distance. Since each vertex always has distance 1 from its neighbours, the fact that the average distance between its neighbours is different from 1 reminds what happens when one measures segments on a curved surface. The graph can be embedded in a geometric space. Communities appear as portions of the graph with a large curvature. The algorithm was applied to the graph representation of the World Wide Web, where vertices are web pages and edges are the hyperlinks that take users from a page to the other. The authors found that communities correspond to web pages dealing with the same topic. Zhang et al [ZWL$^+$08a] have rewritten the clustering coefficient using the fraction of cycles with size four in bipartite networks. They have defined an edge-clustering coefficient of bipartite networks to detect the community structure in original bipartite networks.

4.4.6 Algorithm based on local properties

Bagrow et al [BB05] have presented a detection community method which use local information. Communities are defined locally, based on a simple criterion involving the number of edges inside and outside a group of vertices. One starts from a vertex-origin and keeps adding vertices lying on successive shells, where a shell is defined as a set of vertices at a fixed geodesic distance from the origin. The first shell includes the nearest neighbours of the origin, the second the next-to-nearest neighbours, and so on. At each iteration, one calculates the number of edges connecting vertices of the new layer to vertices inside and outside the running cluster. If the ratio of these two numbers (emerging degree) exceeds some predefined threshold, the vertices of the new shell are added to the cluster, otherwise the process stops. However, the detected communities depends closely on the localization of the starting node and the algorithm requires a time $O(n^3)$.

4.4.7 Algorithm based on global properties

An iterative community detection algorithm based on a measure of information discrepancy (MID) have been proposed by Zhang et al. [ZZZ08]. The authors have defined the profile of each node based on the shortest path (SP) between all the other nodes in the network; if the profiles of a set of nodes are similar, then they can be in the same community, due to the fact that the profile of each node characterizes its overall connection information in the whole network. Thus, if two nodes i and j have similar SP profiles, they must have a very close relationship. In this algorithm, the nodes which have a larger degree and have a less probabilities than a predetermined value are called hubs, when nodes are closer to one hub they form a hub community. An iterative procedure is used to identify hub community and non hub community.

4.5 Divisive methods

The divisive methods split the graph into several communities by removing gradually the edges which connect two distinct communities. Thus, the network is divided into several components representing communities. This removal process of edges can be stopped at any step according to constraint used in each divisive method.

4.5.1 Random walk techniques

Zhou et al. [Zho03a] [Zho03b] have proposed a divisive algorithm based on dissimilarity index between nearest neighbors of a network. Initially, the approach [Zho03b] considers the whole network as a single community. For each community, a threshold parameter θ is computed and equals the initial value θ_{upp} (the upper dissimilarity threshold) of that community. Repetitively, the algorithm carries out changes of the dissimilarity threshold and finds out the nearest neighbors which are friends (if $\Lambda_{ij} <= \theta$, then nodes i and j are friends). Different sets of friends are then formed. A node that does not have any friends is moved to the set of friends in which it has the strongest interaction. In each phase, the nodes are distributed into a number of disjoint communities. The obtained dendrogram shows the relationship between communities and gives the upper and lower dissimilarity thresholds of each community.

4.5.2 Algorithms based on modularity optimization

Several works have used approximation algorithms and heuristics to deal with the modularity optimization problem [BDG+06]. Duch et al [DA05] have proposed an heuristic search method to optimize the search space and find the optimal modularity. They have defined the total modularity Q as the sum of the local modularity q_i on each node such as q_i is normalized in the interval $[-1, 1]$ and is written as

$$\lambda_i = \frac{q_i}{k_i} = \frac{k_r(i)}{k_i} - a_r(i) \tag{4.9}$$

Where: λ_i is the fitness of node i, k_i is the degree of node i, and $k_r(i)$ is the number of links that a node i belonging to a community r has with nodes into the same community. Initially, the algorithm splits the nodes of the whole graph in two random partitions having the same number of nodes each one. This initial communities are considered as connected components for each partition. At each time step, the system self-organizes moving the node with the lower fitness (extremal) from one partition to an other one. In principle, each movement implies the recalculation of the fitness function of many nodes. The process is repeated until an optimal state with a maximum value of Q is reached. After that, the algorithm deletes all the links between both partitions and proceed recursively with every resultant connected component. The process finishes when the modularity Q could not be improved. The algorithm runs in $O(n^2 logn)$, however, the final partition depends closely on the initialization phase of the random partition in the network.

Newman et al [NL07] have used the probabilistic mixture models and the expectation maximization algorithm to infer module assignments and to identify the optimal number of modules in a complex network. The method divides nodes into classes such that the members of each class have a similar patterns of connection. Expectation maximization algorithms are known to converge to local maxima of the likelihood but not always to global maxima, and hence it is possible to get different solutions from different starting points.

Ball et al. [BKN11] have proposed a method for finding overlapping communities based on a generative model of links. The kind of a link presents a main factor to determine all communities. The expected number of links of color z that lies between nodes i and j is $\theta_{iz}\theta_{jz}$ (or $\frac{1}{2\theta_{iz}\theta_{jz}}$ in case of self-edges). This approach uses an expectation maximization (EM) algorithm to find a maximum log likelihood. It solves the following equations:

$$\theta_{iz} = \frac{\sum_j A_{ij} q_{ij}(z)}{\sqrt{\sum_{ij} A_{ij} q_{ij}(z)}} \tag{4.10}$$

Where $q_{ij}(z)$ is the probability that an edge between i and j has color z, which is the quantity needed in order to infer link communities in the network.

$$q_{ij}(z) = \frac{\theta_{iz}\theta_{jz}}{\sum_z \theta_{iz}\theta_{jz}} \tag{4.11}$$

Thus, the final algorithm for dividing the network iterates the EM equations to convergence and, then, assigns each node to the community for which $\theta_{iz}^{(i)}$ is the largest.

4.5.3 Algorithms based on spectral analysis

Newman and Leicht [LN08] have proposed an extension of the spectral optimization method and have rewritten the modularity [New06] for the directed complex networks: If an edge is directed, the probability that it will be oriented in either of the two possible directions depends on the in- and out-degrees of the end vertices. For instance, taken two vertices i and j, where i (j) has a high (low) indegree and low (high) outdegree, in the null model of modularity an edge will be much more likely to point from j to i than from i to j. Therefore, this proposal defines the modularity as follows:

$$Q = \frac{1}{m} \sum_{ij} [A_{ij} - \frac{k_i^{in} k_j^{out}}{m}] \delta_{c_i, c_j} \tag{4.12}$$

The elements of the modularity matrix are defined as:

$$B_{ij} = A_{ij} - \frac{k_i^{in} k_j^{out}}{m} \tag{4.13}$$

To obtain the symmetric modularity matrix, Q have been rewritten as:

$$Q = \frac{1}{4m} s^T (B + B^T) s = \beta_i (v_i^T s)^2 \tag{4.14}$$

Where: β_i is the eigenvalue of $(B + B^T)$ corresponding to eigen vector v_i and s_i equal to $+1$ if node i is assigned to community 1 and -1 if it is assigned to community 2. Therefore, the algorithm calculates the eigen vector corresponding to the largest positive eigenvalue of the symmetric modularity matrix $(B + B^T)$ to divide the network into two communities, which are identified according to the signs of the elements of the eigen vector. Thus, a generalization of the modularity matrix is given:

$$B_{ij}^{(g)} = B_{ij} - \delta_{ij} \sum_{k \in g} B_{ik} \tag{4.15}$$

Where: $B^{(g)}$ is the modularity matrix of the subgraph. These spectral optimization method extracts directed information of links resulting in identifying a significant community structure in time $O(n^2 logn)$.

In [NN12], the authors focus on the spectral properties of the adjacency and modularity matrices using random matrix methods. This approach is built based on the stochastic block model. The probability of an edge between two nodes depends only on the groups with which the nodes fall. If the diagonal elements of the matrix of probabilities are greater than the off-diagonal elements, then the network displays a community structure with a greater density of edges within groups than between them.

4.5.4 Algorithm based on local properties

Shen et al. [YS] have proposed a filtration recursive method using a random model for networks in order to simultaneously run the removal of several edges in each filtration operation. The algorithm applies a recursive community coefficient (CRC) to quantify the quality of division. The authors have proposed a quality function that recursively optimize the recursive community coefficient. The method [YS] runs in $O(m^2 + (c + 1)m)$, for a network of m edges and c communities.

4.5.5 Algorithm based on global properties

The approaches proposed by Newman et al. [GN02] [NG04] are inspired from Freeman works [Fre77]. The intuitive design of a central point in the communication, which based on the structural property of betweenness, allows to define this point as being connecting between other points along their shortest paths of communication [Fre77]. Edge betweenness is the number of shortest paths between all vertex pairs that run along the edge [New10]. It is an extension to edges of the popular concept of site betweenness, introduced by Freeman in 1977.

Newman et al. [NG04] have defined three measures: shortest-path betweenness, current-flow betweenness and random walk betweenness. The algorithm can calculate the shortest paths between a particular pair of nodes using the breadth-first search in time $O(mn^2)$ [AMO93], [THCS01]. Newman have proposed [New01] a powerful algorithm which find all edge betweennesses in time $O(mn)$. After that, the algorithm detect communities by the removal of the edges with largest betweenness. The algorithm results in high quality partitions for networks of small size. However, at each step, when removing an edge, both of the shortest path betweeness algorithm and random walk betweeness algorithm update all computations which is very expensive in computation and need to be carried out in $O(n^3)$.

Sales-Pardo et al [SPGMA07] have observed that the hierarchical structure gives a very significant knowledge of the dynamics of several complex networks such biological networks [[Bi] [IS05]. It determines the organization of complex systems and extracts the relevant information at each level. They have proposed a top-down method consists of two steps: 1) measuring the similarity between vertices; 2) deriving the hierarchical structure of the graph from the similarity matrix [SF82]. The similarity measure, named node affinity, is based on Newman-Girvan modularity. Basically, the affinity between two vertices is the frequency with which they coexist in the same community in partitions corresponding to local optima of modularity. The latter are configurations for which modularity is stable, i. e. it cannot increase if one shifts one vertex from one cluster to another or by merging or splitting clusters. The algorithm gives meaningful partitions for some social, technological and biological networks but it is quite slow.

Fortunato et al [FLM04] have suggested a divisive algorithm that uses a centrality measure [LM04] which is based on the concept of efficient propagation of information over the network. The efficiency between two nodes i and j is equal to the inverse of the shortest path length, and the average efficiency of the graph G is defined as the average of the individual efficiency over all $n(n-1)$ ordered pairs of distinct nodes. Information centrality can be used to quantify the importance of an edge of the graph G. The information centrality C_k^I of the edge k is defined as the relative deviation efficiency caused by the removal of an edge from G. The algorithm finds and removes iteratively the edge with the highest information centrality. The drawback of this algorithm is the computational cost.

4.5.6 Algorithm based on clustering coefficient properties

Watts and Strogatz [WS98] have proposed a model of clustering coefficient. The clustering coefficient quantifies how the neighbors of a node are well connected in a network. Radicchi et al [RCC+04] have proposed a divisive algorithm for detecting communities using the concept of edge clustering coefficient. They have defined the edge-clustering coefficient by analogy with the node-clustering coefficient. It is the number of triangles to which a given edge belongs divided by the number of triangles that might potentially include it (see figure 4.6).

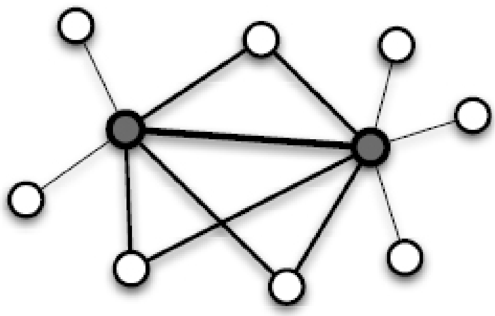


Figure 4.6: Schematic illustration of the edge clustering coefficient introduced by Radicchi et al. The two grey vertices have five and six other neighbors, respectively. Of the five possible triangles based on the edge connecting the grey vertices, three are actually there, yielding an edge clustering coefficient $C^3 = \frac{3}{5}$

Formally, for the edge-connecting node i to node j, the edge-clustering coefficient is defined as:

$$C_{i,j}^{(3)} = \frac{z_{i,j}^{(3)} + 1}{min[(k_i - 1), (k_j - 1)]} \tag{4.16}$$

Where: $z_{ij}^{(3)}$ is the number of triangles built on that edge (i, j) and $min[(k_i - 1), (k_j - 1)]$ is the maximal possible number of these triangles. The idea behind the use of this quantity in a divisive algorithm is that edges connecting nodes in different communities are included in few or no triangles and tend to have small values of $C_{i,j}^{(3)}$. On the other hand, many triangles exist within clusters. Hence, the coefficient $C_{i,j}^{(3)}$ is a measure of how intercommunitarian a link is. A problem arises when the number of triangles is zero, because $C_{i,j}^{(3)} = 0$ irrespective of k_i and k_j, to To remove this degeneracy in the network structure, Radicchi et al [RCC+04] have defined higher order cycles g as:

$$C^{(g)}_{i,j} = \frac{z^{(g)}_{i,j} + 1}{s^{(g)}_{i,j}} \tag{4.17}$$

Where: $z^{(g)}_{i,j}$ is the number of cyclic structures of order g the edge (i, j) belongs to, and $s^{(g)}_{i,j}$ is the number of all possible cyclic structures of order g that can be built given the degrees of the nodes. The algorithm removes the smallest edge-clustering coefficient at each step and each removal operation requires only a local update of clustering coefficients, so the algorithm is much faster than several algorithms.

Lind et al [LGH05] have studied the clustering coefficient of bipartite networks for which there are no cycles of size three, and therefore, the standard definition of clustering coefficient given in [LP49] can not be used. Thus, the coefficient is defined as the number of existing square over the total number of all possible squares.

Zhang et al. [ZWL+08b] have proposed a community detection algorithm for bipartite networks. The main idea is to remove at each step the edge which have the smallest value of edge-clustering coefficient. The authors have defined the clustering coefficient as the fraction of the number of observed squares to the total number of possible squares in the graphs. The edge-clustering coefficient LC_4 is defined as

$$LC_{4,iX} = \frac{q_{iX}}{(k_i - 1)(k_X - 1) + k_i^{(2)} + k_X^{(2)}} \tag{4.18}$$

Where: q_{iX} is the number of squares to which a given edge l_{iX} belongs.k_i is the degree of node i and $k_i^{(2)}$ is the degree of the second neighbors of node i except the nodes which are the first neighbors of node X. In bipartite networks, a clique is a square. It is the basic unit which gives the relationship of two nodes in the same set. The clustering coefficient LC_3 of edge l_{iX} is defined as:

$$LC_{3,iX} = \frac{1}{k_i + k_X - 2}\left(\sum_{m=2}^{k_X} \frac{t_{mi}}{k_m + k_i - t_{mi}} + \sum_{N=2}^{k_i} \frac{t_{NX}}{k_N + k_X - t_{NX}}\right) \tag{4.19}$$

Where: nodes m and i are from the same set and i and X are not in the same set, t_{mi} is the number of triples which contain node m and i, t_{NX} is the number of triples which contain node N and X. The divisive algorithm , for bipartite structure, reveals a good community structure, however, the number of detected communities has to be determined in advance.

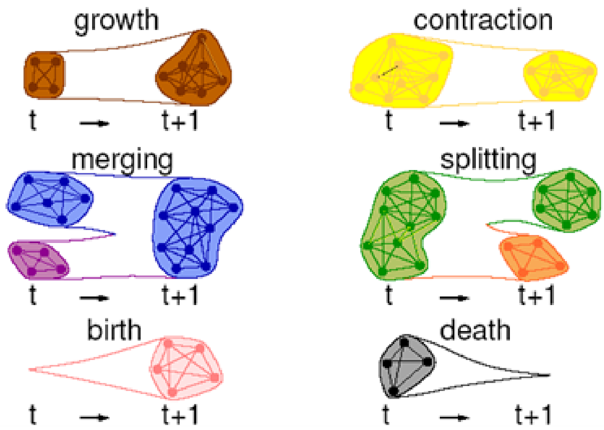


Figure 4.7: Example of a dynamic in communities

4.6 Dynamic community discovery methods

The study of dynamic networks is very interdisciplinary in nature. The network structure, describing how the graph is wired, helps us to understand, predict and optimize the behavior of dynamical systems. In many cases, however, the edges are not continuously active. As an example, in networks of communication via email, text messages, or phone calls, edges represent sequences of instantaneous or practically instantaneous contacts. In some cases, edges are active for non-negligible periods of time: e.g., the proximity patterns of inpatients at hospitals can be represented by a graph where an edge between two individuals is on throughout the time they are at the same ward. Like network topology, the temporal structure of edge activation can affect dynamics of systems interacting through the network, from disease contagion on the network of patients to information diffusion over an e-mail network. The basic community events described in [PBV07] shows the basic events on dynamic communities for complex networks. In figure 4.7, we distinguish between six operations on dynamic communities that are briefly described in the following:

- Birth, when a new community emerges without predecessor.
- Death, when a community disappears without successor.
- Merging, when several communities join together to form a new community.
- Splitting, when a community splits into several new communities.
- Growth, when a community gains new members.
- Contraction, when a community loses members.

Yang et al [YCZ+11] have proposed a dynamic stochastic block model that captures the evolution of communities by explicitly modeling the transition of community memberships for individual nodes in the network. This model assumes that the nodes of the network are divided into classes, and describes the probability of an edge between two nodes based only on the classes to which they belong, it divides the nodes in disjoint classes. The generative process includes the community assignments of all nodes in a social network at a given time step where each node is assigned according to probability π. The links between nodes are generated according to the Bernoulli distribution. A transition matrix A is determined to capture the evolution of communities using an expectation maximization algorithm.
Palla et al [[Pa] have proposed a percolation process to detect the overlapped communities in the networks. The clique percolation method (CPM) is based on first locating all cliques (maximal complete subgraphs) of the network and then identifying the communities by carrying out a standard component analysis of the clique-clique overlap matrix. To detect dynamic communities, this method builds a graph union of the two corresponding networks at two instants t and $t+1$. The authors have determined a correlation function to quantify the overlap between communities over consecutive steps, it is defined as:

$$C_A(t) = \frac{|A(t_0) \bigcap A(t_0+t)|}{|A(t_0) \bigcup A(t_0+t)|} \tag{4.20}$$

However, CPM method supposes that the graph has a large number of cliques, thus it can fail to detect significant partitions for graphs containing just a few cliques.

Greene et al. [GDC10] have defined a dynamic network as a set of l time step graphs $g_1, ..., g_l$, providing snapshots of the nodes and edges in the overall network at successive intervals. Then, authors deal with the identification of a set of $k\prime$ dynamic communities $D = D_1, ..., D_{k\prime}$ that are existing in the network across multiple time steps. The observations can be taken from any disjoint or overlapping groups in order to provide assignments for some or all of the nodes in the complete network. This approach considers this problem as a weighted bipartite matching task. It uses Jaccard coefficient to compare a community over time t and the the recent observation. The algorithm of Bit array operations [APU09] is used for comparison.

Mitra et al.[MTR12] have analyzed communities based on repeated inter-temporal interactions. In dynamic communities nodes switching from a community to another across time, or the possibility that a community survives while its members are being integrally replaced over a longer time period. The authors have proposed a formalism to track the evolution of communities in the context of time-directed datasets such as citation networks.
In the work [AG10], the authors have proposed a method to detect only one decomposition in communities that is good for every time step. The have shown that this unique partition can

be computed with a modification of the Louvain method and that the loss of quality at each time step is generally low despite the constraint of global maximization.

In [NDXT11], the authors have proposed a Quick Community Adaptation (QCA) to discover community structure and track the network changes. They have defined the adaptive community structure detection algorithm as a core for dynamic MANETs based on the idea that Mobile Ad hoc Networks exhibit the existence of groups of nodes where each group is densely connected inside than outside, therefore this method provides speedup and robustness to routing strategies.
We recommend the published paper of Fortunato [For10] for further reading.

4.7 Conclusion

The detection of a significant community structure is an important mechanism for understanding the structures and functions of complex networks. For this reason, several meaningful methods have been proposed in the literature. In this chapter, we have described the principle definitions of community structure, have reviewed the community discovery methods,and, have given keys concepts for further chapters.

CHAPTER 5

AN AD HOC TOPOLOGY SCHEME FOUNDED ON COMMUNITY STRUCTURE PROPRETY

5.1 Introduction

A mobile ad hoc network is a collection of autonomous wireless mobile units that move freely using their wireless interfaces for communication without the aid of an existing infrastructure [Moh02]. Therefore, the mobile network doesn't require a centralized administration entity to manage the operation of the different mobile nodes. Mobile devices profiling aims to take advantage from social proprieties of individuals in order to deploy a new social aware methods for wireless network management. In this work, we seek to construct an efficient topology that realistically establishes communication links with respect to the belonging of the mobile users (their communities). Constructing an efficient topology is important to improve network throughput and conserve energy of mobile devices.
As we discussed in chapter 3, there are two types of topologies [MH]; Covering node Sets (MPR, k-CDS, DS / IS...) and Link Based Structure (DAGs, cliques...). Connected dominated set topologies CDS can be used for building clusters. Thus, CDSs provide several advantages in network applications such as ease of dissemination and of virtual backbone construction [SSZ02], however, connected dominated set topology may generate an undesirable number of cluster heads. Hence, several studies focus on the decision problem of the minimum connected dominating set. Guha et al. [GK98] focus on the question of finding a connected dominating set of minimum size, where the graph induced by vertices in the dominating set is required to be connected as well. In addition, weakly-connected dominating Set (WCDs) have been used to minimize the number of cluster heads [CL03]. As it is NP-complete to determine whether a given graph has a weakly-connected dominating Set of a particular size, the authors present a zonal distributed algorithm for finding small weakly-connected dominating sets. Besides,

several methods have been proposed to construct a Minimum independent set of a graph. We can find a comprehensive survey of the available methods to construct MIS and CDS solutions in ad hoc networks in the work [LWG10]. Several methods of Link Based Structure have been proposed to better construct virtual dynamic topology for routing, data aggregation , ect in ad hoc networks. Thus, in this chapter, we present an efficient distributed topology based on the concept idea of the CDS and the spanning tree [DBS14] to provide a self organized policy to mobile devices with respect to the common interest of the mobile users. We will describe the topology construction phases and will illustrate who each mobile entity is aware about its neighbors which resulted in an efficient topology for communication. The chapter is followed by studying the interaction between virtual topology layer and the routing layer introducing a new topology scheme for a routing application.

5.2 Constructing topology based on structural equivalence

We introduce a novel approach that allow mobile device to reach all members of its community based on an efficient virtual backbone of in-demand community. In fact, we have proposed a distributed algorithm to construct a clustering tree topology based on structural equivalence. In the first phase, the algorithm identifies a sets of nodes that cover all nodes of the graph, regardless of their belonging to communities in order to build a set of privileged nodes and their clusters. In the second phase, the algorithm connects only nodes in the clustering tree to nodes that are either in the same community or they are covering at least one node of this community by creating virtual backbone of community in order to ensure the communication between nodes within a community and manage efficiently the routing requests.

5.2.1 Preliminary and definitions

Let's consider an ad hoc wireless network presented by a graph $G = (V, E)$, with V is the set of nodes and $E \subseteq V^2$ is the set of communication edges, according to the unit disk graph (UDG). We suppose that nodes have the same range transmission and the radio signal propagates according to the free space model. We define
n: the number of nodes in the network, $n = |V|$
k_u: the degree of node u.
Γ: the maximum degree of graph.
$N(u)$: the set of neighbors of node u, $N(u) = \{v \in V/v \neq u \wedge (u, v) \in E\}$
$A_{u,v}$: is the adjacency matrix of the network . It is defined as 1 if nodes u and v are connected, otherwise it is 0.

Each node u belongs to a community of interest denoted by C_u. The goal of this work is to enable a node to reach all members of its community through an optimal route of the graph of communication. Thereby, the proposed algorithm computes similarity between nodes using the information contained in the network structure in order to design an efficient backbone and ensure sufficient coverage of nodes that may participate at the same community. Similarity can be calculated with many different ways. Here, we focus on the topological structure measures [LW71].
Two nodes are structurally equivalent if they share many of the same network neighbors[New10]. The point is to take advantage of the topological perspective of the structural equivalence measure. Intuitively, nodes that have high structural weight maintain high coverage of nodes in the network and nodes that are members in the same community have high probability to have a common neighbors , thus, we will be able to design an efficient backbone of all nodes that belong to the same community. Therefore, we define the structural equivalence:

Definition 11 *Two nodes are structurally equivalent , in a network, if they are connected and share many neighbors.*

Formally, the structurally equivalent between two connected nodes u and v can be written as:

$$\sigma_{vu} = \frac{\sum(A_{u,} * A_{,v}) * A_{uv}}{\sqrt{k_u k_v}} A_{u,v} \tag{5.1}$$

Thus, we define the structural weight for each node:

$$S_u = \sum_v \sigma_{uv} \tag{5.2}$$

Figure. 5.1 shows an example of structural equivalence between two nodes u and v. They have a direct wireless link and share two same neighbors.

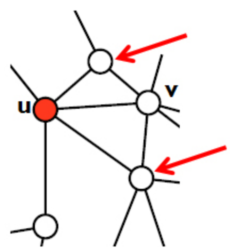


Figure 5.1: Structural equivalence between two nodes.

5.2.2 Neighbors detection

We assume that node u send a message to node v , with respect to direct transmission, the minimum power required to send its message is written as follows:

$$P_t(u,v) = \tau d_{u,v}^{\alpha} \tag{5.3}$$

Where:
$\tau d_{u,v}^{\alpha}$ is the power transmission from node u to node v ,α is the path loss exponent,($2 \leq \alpha \leq 4$), it depends on the characteristics of communication medium [R+96]. We assume that all nodes have the same maximum transmit power and the wireless medium is symmetric. First, the proposed algorithm identifies the list of neighbors with minimum transmit power. Initially, each node u send a control message at maximum power, upon receiving the beacon from a neighbor v, node u can compute the transmit power needed to reach v by comparing the received power of the control message with the maximum transmit power. Then, the algorithm selects the neighbors of node u with minimum transmit power and eliminates the neighbors which requires a transmit power upper than a given threshold of transmit signal strength in order to reduce the number of coordination and the energy consumption.

5.2.3 Clustering tree Algorithm

In this section, we discuss a clustering mechanism; clustering tree algorithm, that exploits the set of privileged nodes to ensure sufficient coverage. Therefore, the algorithm constructs a virtual topology that chooses a best set of nodes which dominate the remaining nodes with a minimum number of links. After the neighbors detection phase, node u broadcasts locally a message containing its ID and the list of its neighbors, upon receiving this message, each neighbors v checks if it has efficient power link $P_t(u,v)$ between its neighbors. If so, it adds them to the set of its neighbors,then, every node updates the adjacency matrix , computes and broadcasts its structural weight in local way. In the last step, the algorithm constructs the clustering tree, which is a subgraph of G.

In order to perform a cluster formation, each node must have its view of the current cluster configuration, thus, we define the state of nodes as follows:
$state_u = CH$, cluster head (CH).
$state_u = SN$, cluster member (Simple Node).
$state_u = LN$, cluster member (Leaf Node).

Several clustering approaches have been proposed; the Lowest-ID algorithm [GT95] gives to each node a separate ID "identification" and periodically broadcasts a list of its neighbors.

This algorithm has been improved in [CWLG97]. Basuet al [BKL01] introduced a metric for mobility, for mobile ad hoc networks (MANET), based on the ratio between the received power levels of successive transmissions measured at any node from all its neighboring nodes. Thereby, the definition of a cluster should not be defined a prior by some fixed criteria, but should reflect the density of the studied network [MBF04]. In our work, the principle idea consists of selecting nodes that have a highest structural weight, as they are the best candidates to cover a larger number of nodes and construct a clustering tree. Therefore, the cluster head selection phase is as follows: let $v \in N(u)$, each node u compares its structural weight with its 1-neighbors $N(u)$, if node u has the highest structural weight ($S(u) > S(v)$), it will be selected as cluster head $L(u) = u$, otherwise the algorithm chooses a parent of node u with highest structural weight $D(u) = w$. To this end, we have defined three cases to allow node u to join its suitable cluster based on its structural weight:

- If the parent of node u (u's parent) is a cluster head, u will join its parent w ($D(u) = L(u) = w$).

- If u's parent is not a cluster head, node u compares its structural weight with that of its neighbors except its parent w (this set is defined by $M(u) = \{u\} \cup N(u) \cap \{w\}$), in order to join the nearest cluster head, thus, node u selects a new parent, and this process will be iterated until that u will join its cluster head.

- If node u has a low degree ($k_u = 2$) or is a leaf node it will join its parent w, then a branch of the tree is extended till it joins its cluster head.

We remark that the procedure for constructing the topology, $Construct(T, u, w)$ builds a clustering tree gradually in which every node belongs to a cluster and the inter-clusters links allow to exchange message between the nodes.

5.3 On construction of virtual community backbone

In this section, we describe the social-aware algorithm and how it builds a virtual community backbone (see algorithm 2). Discovering this underlying structure help mobile devices to being aware about users mobility.
The steps of the proposed algorithm are: a node s sends a search request (s, C_r) to its members of cluster to discover in-demand community C_r. We assume that a request sent by a node is received correctly in a finished time by all its members of cluster. Due to the fact that the

cluster head are coverage nodes, they serve as a relay, and, consequently, they belong to the backbone. Upon receiving such request by a node u, it sends a request reply indicating its state in the cluster (NS, NF).

For a node u that is a simple node, the steps of the algorithm are:

- if a node u belongs to the in-demand community, it responds by sending a request reply $RRep(u, M = 1, Mem = \{(u, L(u))\}, LB = \{\})$ to $D(u)$, where:
  u is the node identifier, the parameter $M = 1$ indicates that node u belongs to the in-demand community , the set $Mem = \{(u, L(u))\}$ contains the cluster head of node u, and the set LB indicates all the nodes that have been identified as a member in the backbone.
- Then, node u joins the local backbone covering by its cluster head $L(u)$, we call a function to build the backbone $LocalBackbone(L(u))$.
- After that, node u will broadcast the search request to the list of its subtrees of children (u 's children) $NC = \{v \in N(u)/D(v) = u\}$ even if it does not belong to the in-demand community in order to find the backbone members by sending $Req(u, C_r)$ to the list $NC(u)$.
- Let v be in the list of subtree NC, node u processes all received requests $RRep(v, M(v), Mem, LB)$ from a node $v \in NC(u)$, so, the algorithm will join node v to the local backbone if $M(v) = 1$.
- If a leaf node u belongs to the in-demand community, it will send the responds to its parent.

5.4 Results

The proposed approach builds an efficient backbone for the users of ad hoc networks with respect to their communities of interest. We have first described a clustering topology phase, where all the nodes get their neighborhood information, update their own data structure (adjacency matrix, structural equivalence, structural weight), and , then, construct a virtual backbone that ensures connectivity and results in a good coverage of members of communities.

```
Data: G(E,V), P_max the maximum node transmit power.
1. Initialization
begin
    N(u) = {}; D(u) = {}; L(u) = {}; M(u) = {};
end
2. Information exchange
begin
    u sends beacon message at transmit power P_max
end
3. Neighbors detection
begin
    upon receiving message from node v
    Compute the P_t(u,v)
    if P_t(u,v) < P_threshold(u,N(u)) then
        N(u) = N(u) ∪ {v}
    end
end
4. Constructing Clustering tree
begin
    u broadcasts locally the list of its neighbor
    Wait for stabilization time
    Update A_uv
    Compute structural equivalence neighbors list(u)
    Node u broadcast its structural weight in beacon message
    while L(u) = {} do
        if (∀v ∈ N(u), S(v) < S(u)) then
            L(u) = u
            Statut_u = CH
            Construct(T, (u, N(u)))
        end
        else
            (∃w ∈ N(u), (∀v ∈ {u} ∪ N(u), S(v) < S(w))
            D(u) = {w}
            L(u) = L(w)
            if D(w) = L(w) = w then
                Statut_u = SN
                if deg(u) = 1 then
                    Statut_u = LN
                end
                Construct(T, u, w)
            end
            else
                begin
                    if D(w) = L(w) ≠ w and deg(u) >= 3 then
                        M(u) = u ∪ N(u) ∩ w
                        order(list(M(u)))
                        (∃z ∈ M(u), (z = first(list(M(u)))
                        if u=z then
                            L(u) = u
                            Statut_u = CH
                            Construct(T, u, M(u))
                        end
                        else
                            D(z) = x, x ∈ N(z)
                            L(u) = L(z) = L(x) Iterated until that u is joined to its cluster head
                        end
                    end
                    else
                        if deg(u) < 3 then
                            D(w) = k, k ∈ N(w)
                            Statut_u = SN
                            if deg(u) = 1 then
                                Statut_u = LN
                            end
                            Construct(T, u, w)
                            L(u) = L(w) = L(k) Iterated until that u is joined to its cluster head
                        end
                    end
                end
            end
```

Data: $T, L(u), D(u)$
Procedure CommunityDiscoveryBackbobne ()
begin
    M: indicate if a node belongs to the requested community.
    $Cache$: caching of node u
    $LB = \{\}$
    Upon receiving the request packet $Req(s, C_r)$ from u node
    if $(statut_u = NS)$ **then**
        if $(C_u = C_r)$ **then**
            Send $RRep(u, M = 1, Mem = \{(u, L(u))\}, LB = \{u\})$ to $D(u)$
            Add u to $LocalBackbone(L(u))$
        end
        $NC = \{v \in N(u)/D(v) = u\}$
        Send Request packet $Req(u, C_r)$ to $NC(u)$
        time stabilization
        for *each received* $RRep(v, M(v), Mem, LB)$ *from* $v \in NC(u)$ **do**
            if $(M(v) = 1)$ **then**
                $LB = LB \cup \{v \in NC(u)/M(v) = 1 \wedge D(v) = u\}$
                $LocalBackbone = true$
            end
        end
        if $(LocalBackbone = true) and (C_u = C_r)$ **then**
            Send $RRep(u, M = 1, Mem = \{(u, L(u))\}, LB)$ to $L(u)$
            Add all nodes in LB to LocalBackbone(L(u))
        end
        if $(C_u \neq C_r) and (LocalBackbone = true)$ **then**
            Send $RRep(u, M = 0, Mem = \{(u, L(u))\}, LB)$ to $L(u)$
            Add u to $LocalBackbone(L(u))$
            Add all nodes in LB to $LocalBackbone(L(u))$
        end
    end
    else
        if $(statut_u = NF)$ **then**
            if $(C_u = C_r)$ **then**
                Send $RRep(u, M = 1, Mem = \{(u, L(u))\}, LB = \phi)$ to $D(u)$
                Add u to Local$Backbone(L(u))$
            end
        end
    end
end
begin
    Broadcast $Req(s, C_r)$ to node member of cluster for 1 hop
    $CommunityDiscoveryBackbobne()$
    Send information for Update cache of node s
end
return $GlobalBackbone(C_r)$

Algorithm 2: Constructing Global Backbone Algorithm - phase 2

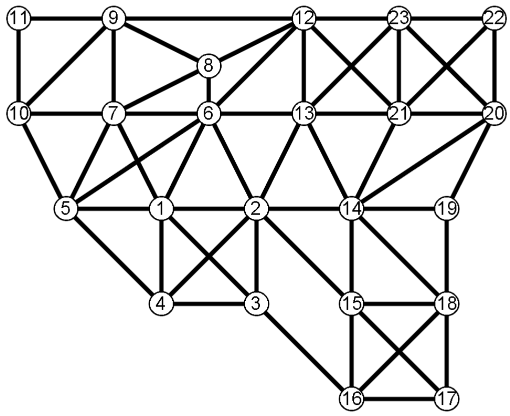


Figure 5.2: Initial network

In Figure. 5.2, the network contains 23 nodes and 54 links. First, the algorithm eliminates the non-efficient power links. Then, it computes the structural weight of each node. Table 5.1 gives the structural equivalence between nodes and the structural weight of each node. The algorithm selects the leaders 1, 9,15, and 23 because they have the highest structural weight 2.19, 1.63, 1.64, and 1.83 respectively. During the construction of clusters the algorithm selects the inter-cluster links with a strength connectivity link (red lines in figure 5.3). The result is a clustering tree topology as is illustrated in figure 5.3(a).

The second phase of the algorithm selects the members of the virtual backbone. Figure 5.3(b). shows an example about how to discover a virtual community backbone (the backbone of community (1) is presented by red line), for example, node s ($id = 8$) sends a request for discovering its community members, upon receiving this request from the cluster head 9, node s will broadcasts the search request to its cluster members, therefore, if the leaf nodes 10 and 11 belong to the in-demand community $C_r = 1$, they will send a reply to node 9 and so on. The clustering tree algorithms generate several messages overhead. We compute the total of the all exchanged message; when all nodes are looking at their community members by sending a request to their cluster head, so, in this case, the message complexity is $O(n\Gamma^2)$, where Γ is the maximum degree of graph. As a result of our approach, the topology are planar and ensures connectivity.

Table 5.1: Structural equivalence and structural weight of the studied network.

Nodes	$\sigma_{i,j}$						S
1	$\sigma_{1,2(0,40)}$	$\sigma_{1,3(0,47)}$	$\sigma_{1,4(0,47)}$	$\sigma_{1,5(0,47)}$	$\sigma_{1,6(0,18)}$	$\sigma_{1,7(0,18)}$	2, 19
2	$\sigma_{2,1(0,40)}$	$\sigma_{2,3(0,28)}$	$\sigma_{2,6(0,22)}$				0, 90
3	$\sigma_{3,1(0,47)}$	$\sigma_{3,2(0,28)}$	$\sigma_{3,4(0,33)}$				1, 08
4	$\sigma_{4,1(0,47)}$	$\sigma_{4,3(0,33)}$	$\sigma_{4,5(0,33)}$				1, 13
5	$\sigma_{5,1(0,47)}$	$\sigma_{5,4(0,33})$	$\sigma_{5,7(0,26)}$				1, 06
6	$\sigma_{6,1(0,18)}$	$\sigma_{6,2(0,22)}$	$\sigma_{6,8(0,22)}$	$\sigma_{6,12(0,45)}$	$\sigma_{6,13(0,20)}$		1, 28
7	$\sigma_{7,1(0,18)}$	$\sigma_{7,5(0,26)}$	$\sigma_{7,8(0,22)}$	$\sigma_{7,9(0,45)}$	$\sigma_{7,10(0,26)}$		1, 37
8	$\sigma_{8,6(0,22)}$	$\sigma_{8,7(0,22)}$	$\sigma_{8,9(0,25)}$	$\sigma_{8,12(0,25)}$			0, 95
9	$\sigma_{9,7(0,45)}$	$\sigma_{9,8(0,25)}$	$\sigma_{9,10(0,(58)}$	$\sigma_{9,11(0,35)}$			1, 63
10	$\sigma_{10,7(0,26)}$	$\sigma_{10,9(0,58)}$	$\sigma_{10,11(0,41)}$				1, 24
11	$\sigma_{11,9(0,35)}$	$\sigma_{11,10(0,41)}$					0, 76
12	$\sigma_{12,6(0,45)}$	$\sigma_{12,8(0,25)}$	$\sigma_{12,13(0,5)}$	$\sigma_{12,23(0,22)}$			1, 37
13	$\sigma_{13,6(0,20)}$	$\sigma_{13,12(0,45)}$	$\sigma_{13,14(0.18)}$	$\sigma_{13,21(0,45)}$	$\sigma_{13,23(0,40)}$		1, 68
14	$\sigma_{13,14(0.18)}$	$\sigma_{14,15(0,20)}$	$\sigma_{14,18(0,41)}$	$\sigma_{14,19(0,24)}$	$\sigma_{14,21(0.20)}$		1.23
15	$\sigma_{15,14(0,20)}$	$\sigma_{15,16(0,35)}$	$\sigma_{15,17(0,58)}$	$\sigma_{15,18(0,50)}$			1, 64
16	$\sigma_{16,15(0,35)}$	$\sigma_{16,17(0.41)}$					0, 75
17	$\sigma_{17,15(0,58)}$	$\sigma_{17,16(0,41)}$	$\sigma_{17,18(0,29)}$				1, 27
18	$\sigma_{18,14(0,41)}$	$\sigma_{18,15(0,50)}$	$\sigma_{18,17(0,29)}$	$\sigma_{18,19(0,29)}$			1, 49
19	$\sigma_{19,14(0,24)}$	$\sigma_{19,18(0,29)}$					0, 52
20	$\sigma_{20,21(0,25)}$	$\sigma_{20,22(0,35)}$	$\sigma_{20,23(0,45)}$				1, 05
21	$\sigma_{21,13(0,45)}$	$\sigma_{14,21(0.20)}$	$\sigma_{21,20(0,25)}$	$\sigma_{21,23(0,45)}$			1, 35
22	$\sigma_{22,20(0,35)}$	$\sigma_{22,23(0,32)}$					0, 67
23	$\sigma_{23,12(0,22)}$	$\sigma_{23,13(0,40)}$	$\sigma_{23,20(0,45)}$	$\sigma_{23,21(0,45)}$	$\sigma_{23,22(0,32)}$		1, 83

We have built our own simulator , in C++, to evaluate the performances of our scheme and compare it with the results given when using MIS backbone algorithm [GHJS03]. We have studied the quality of this proposed algorithm for finding efficient connected covering sets, measuring the backbone nodes set size, and the network life time. We compare our approach to the Minimum Independent Dominating Set algorithm proposed, and we have added to this scheme a clustering steps to exchanging messages between each MIS nodes and its dominated nodes. In our experiments, we generate random graphs repeatedly and run the two methods. The size of the graphs ranges from 50 to 100 nodes, and to simulate the structure of ad hoc networks, all nodes have the same transmission range and all nodes are deployed in a square area.

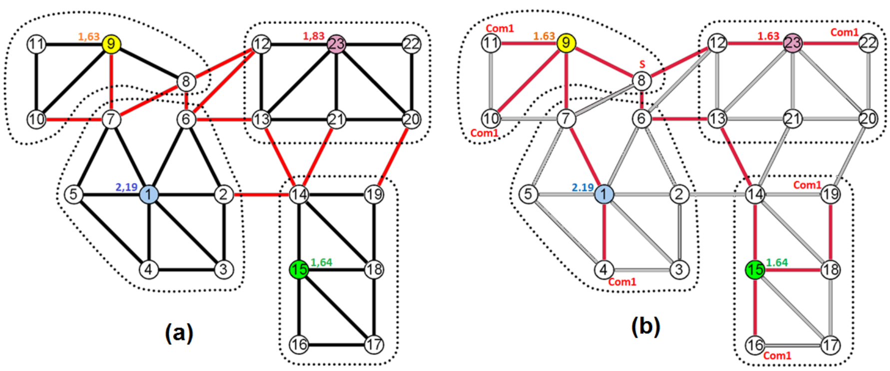


Figure 5.3: An example illustrates the network topology:(a) a clustering tree network topology .(b)the community backbone.

As we can see from figure 5.4 (a), the size of backbone nodes of the in-demand community C_r increases as the network size increases but it is smaller than the size of cluster head nodes produced by MIS topology. This is due to the fact that our approach selects the best nodes candidates for cluster head that covering larger number of nodes (cluster members and members of communities). The simulation results show that the awareness of nodes , in our approach, performs better in prolonging network lifetime (see figure 5.4 (b)).

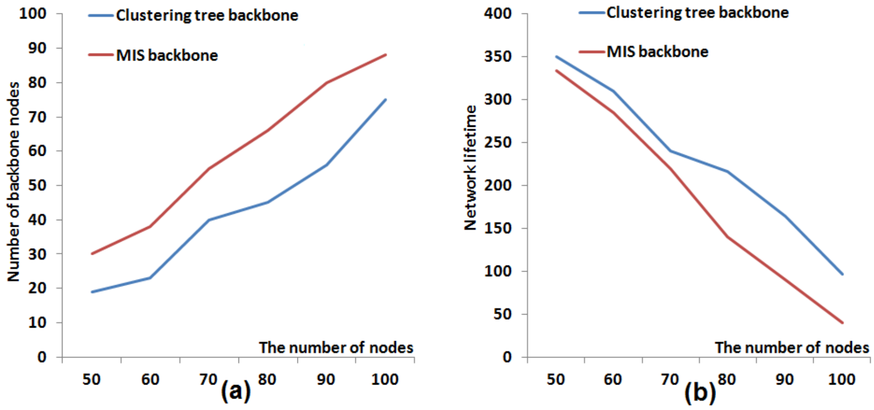


Figure 5.4: Comparaison between MIS backbone and virtual community backbone. (a)The size of backbone nodes .(b) Network lifetime.

5.5 Routing application and topology features

The issues of finding optimal topologies and investigating their features in the layers of the protocol stack such as routing layer has been discussed in several works [San05]. In this section, we aim to understand the interaction between routing layer and topology layer. For this raison, we have proposed an improvement of the DSR protocol based on optimal topology of relay region and enclosure graph, particularly, we discuss the potential advantages of using an optimal topology in wireless networks to reduce node energy consumption [AD].

5.5.1 Key idea

In our work [DB13], we have proposed an extension for DSR protocol using an efficient energy technique. The basic idea is to use the concepts of relay-region and enclosure graph, proposed for the control topology protocol, in the work of Rodoplu et Meng [RM99b]. The authors illustrate that a simple local optimization scheme executed at each node guarantees strong connectivity of the entire network and attains the global minimum energy solution for stationary networks.

5.5.2 Relay region

Let us consider an ad hoc wireless network presented by a graph $G = (V, E)$, with V is the set of nodes and $E \subseteq V^2$ is the set of communication edges, according to the unit disk graph model (UDG).
We suppose that node u must send a packet to node v, which is at distance d (see figure 5.5). Nodes have the same range transmission and the radio signal propagates with the free space model. Thus, the power needed to send a message directly from u to v is proportional to d^2. If node w is in region relay of node u, the distance is given: $d^2 \geq d_1^2 + d_2^2$. In this case, using the intermediate node w to relay u's packet is preferable from the energy consumption's point of view.

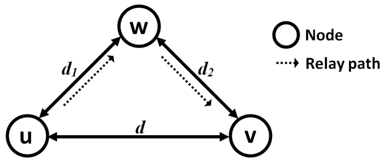


Figure 5.5: Using a relay region: node u must send a packet to v, which is at distance d; using the intermediate node w to relay u's packet is preferable from the energy consumption's point of view.

The region relay of a pair transmitter-relay of node (u, w) identifies the set of points in the plan for which the communication via the node relay is more efficient in energy conservation than a direct communication [RM99b]. Formally, the region relay is defined:

$$RR_{u \to v} = \{(x, y) \in R^2 : P_{u \to w \to (x,y)} < P_{u \to (x,y)}\} \tag{5.4}$$

where
$P_{u \to (x,y)}$: the power required to transmit from node u to node v.
$P_{u \to w \to (x,y)}$: the power required to transmit from node u to node v through an intermediate node w to relay u's packet.

The enclosure of node u represents the region of the plane beyond which it is not energy efficient for node u to search for one hop neighbors.

5.5.3 Energy consumption model

We have discussed the available energy consumption models in [Dri], most of them including transmitted power, received power and idle power. We suppose that node u wants to send a message to node v , in case of direct transmission, the minimum power required to send the message is written as follows:

$$P_{u \to v} = \tau d_{u,v}^{\alpha} \tag{5.5}$$

Where:

$\tau d_{u,v}^{\alpha}$: is the energy transmission from node u to node v, α is the path loss exponent,$2 \leq \alpha \leq 4$, it depends on the characteristics of communication medium [R+96].
If node w is a relay node, the total transmit power is:

$$P_{u \to w \to v} = P_{u \to w} + P_{w \to v} + c \tag{5.6}$$

Where:
c: is a constant term that accounts for the receiver power consumed at the relay node w.
The residential energy of a node u at time t is given in equation (5.7).

$$E_u(t) = E_u(current) - [P_{tx} + C_{rx} + I_{idle}] \tag{5.7}$$

Where:
$E_u(current)$: is the current energy of the battery.
P_{tx}: is the total energy consumed by node u during packet transmission.
C_{rx}: is the total energy consumed at node u during packet reception.
I_{idle}: represents the total energy consumed by each node when the state of the wireless channel is idle.

5.5.4 Energy Aware Route Discovery Process

The Power Control Dynamic Source Routing algorithm (PCDSR) aims to minimize the power consumption during the route construction phase and routing process. The structure of the route request packet of DSR protocol has been modified to carry information of power level and it will be updated periodically. The energy model takes into account the energy consumed during packets transmission, packets reception, and power consumed in the idle phase. Initially all nodes have the same initial energy. The main steps of the proposed algorithm are presented in algorithm 3.

In PCDSR mechanism, a node computes its relay region in order to relay packets through intermediate nodes instead of using a long energy-inefficient edge. Therefore, nodes consume minimal transmit power while detecting neighboring nodes. Further, the algorithm defines weight to each link, and , upon receiving route reply, it calculates the total costs of the discovered routing paths. Then, source node saves the routes discovered in the route cache and select the best routing path.

```
1. Initialization:
N(u) = ∅ is the neighbor set of node u
N: total number of nodes
d: destination node.
γ(t): A threshold of remaining battery capacity at time t.
2. Forwarding the request packet:
A source node s forwards a Request packet
for each node u ≠ d that have received Request packet do
    Update N(u)
    Update routing Table
    Update the weight field in the request packet weight(k, u)
    let k ∈ N(u)
    Compute remaining power E_u(t)
    E_u(t) = E_u(current) − [P_{k,u} + C_{k,u} + I_idle]
    E_u(current) = E_u(t)
    if (E_u(t) > thresholdvalue) then
        RelayRegion(u)
        if ( ∃ ((k − u) and (k − f − u) in header of request packet) and (P_{k→f→u} < P_{k→u}) then
            Node u ignore request packet with (k − u) path
            u update its routing table and save (k − f − u) path.
        end
        Node u send a Request packet at minimum power Pmin_u
        Compute the minimum weight of link : weight(u, u_{i+1}) = E_u(current) − Pmin_u
    end
end
3. Receive reply packet:
for each path ρ_i do
    for each node u to destination node d do
        Send reply packet
        Compute the path power Cost:
        Costpath_ρ = Σ_{i=0}^{h} weight(u_i, u_{i+1})
    end
end
Select the minimum path:
R = min(Costpath_ρ)
End
```

Algorithm 3: Power Control Dynamic Source Routing algorithm

5.5.5 Results

We simulate our approach in NS-2 [[FV]. We give the following scenario: size of nodes = 20 nodes, the area of simulation is $1000mx1000m$, simulation was performed in $1000seconds$, and we use Random Walk Mobility Model. The propagation model is Two ray ground model. Figure 5.6 illustrates the total energy evolution in time when running the DSR and PCDSR. Network lifetime always play a vital role in determining the network connectivity and network capacity of energy constraints network, we remark that PCDSR gives better network lifetime as compared to DSR.

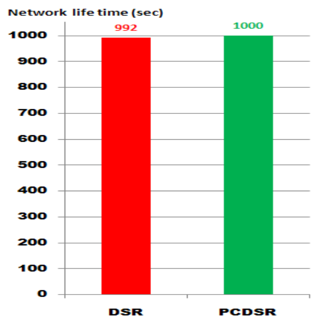


Figure 5.6: Network lifetime

In figure 5.7, we highlight the relative performance of DSR and PCDSR protocols for total received Packet with varying numbers of nodes. We observe that PCDSR protocol has better performance than DSR protocol.

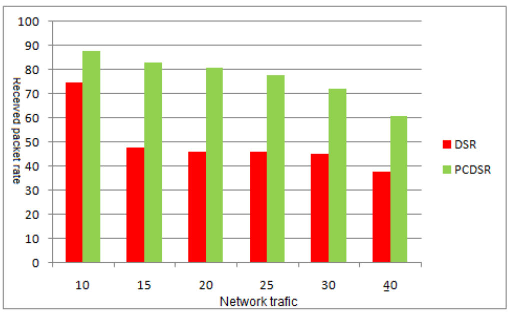


Figure 5.7: Rate of received packets of DSR vs PCDSR

The results show that the PCDSR protocol is more efficient than the DSR protocol in terms of energy consumption. This analysis illustrates that investigating the potential advantages of a virtual topology layer will increase the availability of resources and ensures a better self-organization of mobile devices.

5.6 Conclusion

Communication on a ad hoc wireless channel is characterized by scarce resource availability; in particular, energy and network bandwidth are available in very limited amounts. Therefore, we have proposed a new approach of constructing an efficient topology with respect of the information of communities of mobile users that have been extracted from human mobility analysis to provide the desired performance level and help mobile devices to being aware about users mobility.

Chapter 6

A Pattern Extraction Methodology for Mobility Paths

6.1 Introduction

The growing interest in mobile social networking inspired a lot of efforts for uncovering mobility patterns from various mobility datasets, e.g., individual human mobility data [FHR09] [LZX+08] [CLC10] [RSH+09], vehicle movement data [CMC05], and synthetic mobility data [NP06]. Due to the enormous volume of high resolution positioning data, GPS data in particular, it becomes imperative to developed mining mobility patterns which are useful in many application. Therefore, the extraction of mobility patterns from users travel history has been well studied in the literature and many issues of individuals' mobility have been proposed. User similarity has also been explored for understanding the correlation between user behaviours. In the work [PJZ+16], the authors studied human behaviors in terms of points of interests (PoIs) which a user visits and they found their classes of relevance that indicate significant locations of user during its daily movement. Zheng and al [ZX] proposed an Hypertext Induced Topic Search based inference model (HITS based inference model) to infer the interest level of a location and a users ' travel experience based on GPS trajectories within a region, this model shows that users ' travel experiences and the interests of location have a mutual reinforcement relationship. However, the studying of a group who share a common interest or a set of expected behaviors, with highly users correlation, is a new challenge for mobility data analysis, and it would have a very relevant value to explore the social world around people by mining the communities of mobile users. In this way focus moves from a single users to study of statistical properties of the network as a whole [Newa], [Bar],[Nex]. Furthermore, it is essential to observe the behaviors of users within their communities and analyze them to find the kind and objectives of a community that may indicate the main driver of individual movements. However, this raises several questions: which kind of mobile communities are important for analyzing the users ' mobility? What role plays a common interest in predicting human mobility? How do communities provide accurate knowledge to analyze the different forms of human mobility? We aim to answer to such questions by providing a simple but effective methodology for efficiently identifying mobiles communities based on common interests. Firstly, we detect meaningful stays from individual's traces. Further, these stay points are clustered in order to understand the relationship between the geographic location and the activities and to study the concept of interesting locations for people. Then, we introduce the new concept of *interest based mobile communities*, and apply a community detection method to extract the patterns of connections between users that share a common interest. Finally, we study the community structure and analyze the role of these mobile communities to make evidence of the added value of this novel concept. The discovered communities show that social dimension is related to the contexts in which the human interactions occur. The evaluations and test have been conducted on real dataset.

Therefore, in our work[DBSG17] , we produce the following relevant contributions:

- the introduction of the novel concept of *interest based mobile communities* that allows to derive the following properties:
  - people share common PoIs, but even more they share common interests;
  - people tend to move toward common PoIs over specific period of time while they can share a similar interest regularly or from time to time;
  - we can assign each person to a interest based mobile community;
  - most of the people tend to remain within their interest based mobile community;
- a methodology that allows to extract interest based mobile communities and to provide an *abstract level* for understanding the relationship between people;
- the validation of the relevance of interest based mobile communities that confirms that:
  - interest based mobile communities can provide an effective while simple way for analyzing people mobility;
  - simple classification of interests can suffice to a powerful classification;

The remainder of the our contribution is structured as follows. Section 2 explains the general methodology. We detail stay point extraction in Section 3. In Section 4 we discuss the discovery phase that gives a pattern mining algorithm, computes similarities and uses a community detection method in order to analyze mobile communities. The experiments results and evaluation are described in Section 5. Conclusions and future work are presented in Section 6.

6.2 Proposed Methodology

Contextual dimension of human interaction is increasingly mobilized by the impacts of various mobile technologies and the growth of social networks. Our society is transforming itself into a mobile society, where interaction itself is mobilized[SMK]. That's why, in this work, we are interested in behavioural aspects of individuals to discover groups which have a common interest and have done similar activities according to the context of the visited places.

Further, it is important to determine the type of relation between users within each communities and clarify what is the meaning of the detected communities. Our idea revolves around two observations: 1)The more similar location histories are shared by two users, the more likely the two users belong to the same communities (friends, colleagues, family), 2) The main reason that motivates individuals to choose to visit or attend a commonly frequented public areas or a most interesting locations are their social preferences (social networks, individuals with whom they share a common interest,...). We divide our work in three main steps: stay point extraction, pattern discovery and pattern analysis. The focus is put on taking advantage of user interests and preferences, which is actually more valuable than the interest for a specific location. Furthermore, this contextual view of discovering communities based on user graph provides a higher abstract level that allows to explore the relationship between people. For this reason, we extract the location history of an individual using the individual ' s stay points, and we identify PoIs to mine knowledge about the activities a user does practice within a specific geographic area. Then we derive the similar interests among people at this level. This allows us to further estimate the similarity between different individuals which induces the extraction of their mobile communities patterns. Figure 6.1 recapitulates the structure of our methodology.

6.3 Stay point extraction

In this section, we detect the stay points from a user ' s mobility path by seeking the spatial region where the user stayed for a while. We have applied the algorithm that have been proposed in [LZX+08] in order to extract the stay point that carries more semantic meaning [ZKS].

Definition 12 *A distinct individual 's mobility path* $mp_i = ((p_1, t_1), ..., (p_m, t_m))$ *is a sequence of spatio-temporal points, with* t_i *a timestamp and* $p_i \in R^2$.

We can then define the mobility path:

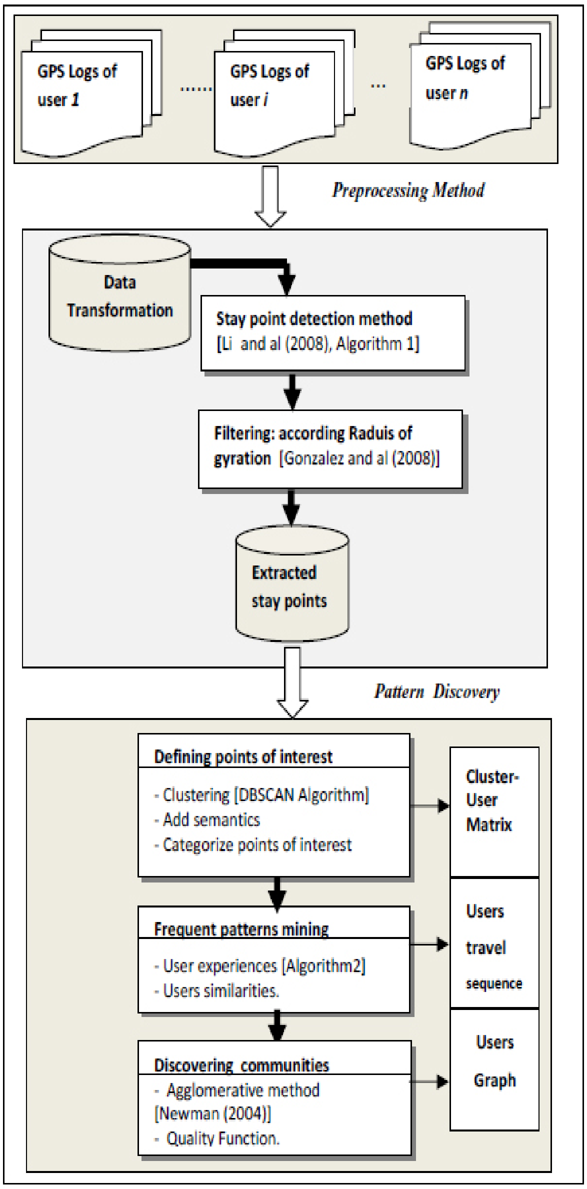


Figure 6.1: The Proposed Methodology for Stay point extraction, Pattern Discovery and Analysis in mobile wireless networks

Definition 13 *A mobility path mp is a sequence of distinct individual 's mobility paths* $mp = \{mp_1, mp_2, ..., mp_n\}$.

We can give the definition of stay point:

Definition 14 *Given a mobility path mp for a given user u, a stay point s is defined according the duration and length of the stay. For each sequence of position points* $P_i = \{p_v, p_{v+1}, ..., p_{v+k}\} \in mp$ *it exists a single stay point* s_i, *if* $\forall v <= v+k <= m$, $Distance(p_v, p_{v+k}) <= D_{max}$ *and* $\Delta t_{(p_v, p_{v+k})} >= T_{max}$, *and we have* $s_i = (M.lat, M.long)$, *where the mean coordinates*

$M.lat$ $M.long$ represent the average latitude and longitude of the collection P.

And further define the set of stay points for a user.

Definition 15 *Then the stay points of a user u_i is the set $S^{(i)}$ such that $S^{(i)} = \{(s_1, arvT_1, levT_1, mp_1), (s_2, arvT_2, levT_2, mp_1), ..., ((s_n, arvT_n, levT_n, mp_n))\}$ where $arvT_k$ and $levT_k$ are the arrival time and the leaving time at s_k.*

The stay point extraction phase is presented in algorithm 1. We consider three aspects when extracting stay points:

- When the threshold distance between two points p_j and p_k on a user's mobility path is greater than D_{max} meters, we assume that this is a significant move to another location that might be a stay point according to the stay duration that a user has spent to.
- Considering the period in which a user stays still in a place assuming the time span between two points p_j and p_k greater or equal to T_{max} minutes, therefore we have obtained the user locations where he spent a significant time, mostly to practice regular activities (work, shopping, sports, restaurant, ..,ect.).
- The algorithm operates on each mobile path separately in order to extract meaningful stays.

Data: D_{max}, T_{max}, $U = \{u_1, u_2, ..., u_{n_u}\}$,$mp = \{mp_1, mp_2, ..., mp_n\}$

forall the *user* $u_i \in U$ **do**
    forall the *mobile paths* $mp_i = ((p_1, t_1), ..., (p_m, t_m))$ **do**
        $find \leftarrow false$
        $k \leftarrow 1$
        while $k <= m$ **do**
            $j \leftarrow k + 1$
            while $j <= m$ **do**
                distance= Dist.twopoints(p_k, p_j)
                30 **if** ($distance > D_{max}$) **then**
                    $\Delta t_{(p_k, p_j)} = t_j - t_k$
                    if $\Delta t_{(p_k, p_j)} >= T_{max}$ **then**
                        $[M.lat, M.long] \leftarrow mean(\{p_l \; k <= l <= j\})$
                        $s_l \leftarrow (M.lat, M.long, alt, arvT_k, levT_j)$
                        Insert.Table($id_s, u_i, mp_i, M.lat, M.long, arvT_k, arvT_j$)
                        $k \leftarrow j$
                        $find \leftarrow true$
                    end
                end
                if *find==true* **then**
                    $find \leftarrow false$
                    break();
                end
                $j \leftarrow j + 1$
            end
            $k \leftarrow k + 1$
        end
    end
    return StayPointTable()
end

Algorithm 4: Stay points extraction

After the stay point extraction, we perform a filtering step to take into account the radius of the representative movement area of a user. It has been observed that users exhibit significantly different mobility behaviours, with respect to their radius of gyration[SQBB]. Therefore, we will compute the radius of gyration, defined as the deviation of user positions from the corresponding centroid position [GHB]. Radius of gyration is defined as follow

$$r_g(t) = \sqrt{\frac{1}{n(t)} \sum_{i=1}^{n} (distance(p_i, p_c))^2} \tag{6.1}$$

where p_i represents a location at time t, and $p_c = \frac{1}{n} \sum_{i=1}^{n} p_i$ is the center of the mobile path. n represents the number of points from the user' s observed mobility path up to time t. $distance()$ is a function calculating the minimum distance over the earth's surface between the locations given in longitude and latitude. Here, we have applied the Vincenty's formulate [Vin]. The individuals who make average or small radius of gyration are ordinary users and thus they have a strong tendency to form communities with their colleagues, friends, family,.. ect, whereas those with large radius of gyration who have high mobility, they participates in many communities. This means that a large radius of gyration provide an efficient way to study the changes that occur in community memberships over time. In this work, we don't consider the growth of communities over time, we leave such analysis for future work.

6.4 Pattern discovery

6.4.1 Defining point of interest

In this phase, we will identify clusters via a density-based clustering algorithm to define a point of interest (PoI). First, a density-based clustering algorithm DBSCAN algorithm [EKS$^+$] is applied on extracted stay points. It require two parameters; ϵ represents the minimum number of neighbors that a node must have to be considered in a cluster, and δ the maximum distance such that two points are considered neighbors (in our tests we put $\epsilon = 7$,$\delta = 10$). As a result, we get a group of users stay points that share common locations or are relatively close to each other. Then, we aim to understand which activities does user practice within different clusters and study how certain places are affected by human systematic mobility by evaluating the visit frequency of PoI and looking at users behavior.
Unlike the work of Papandrea et al[PZG$^+$13], [PJZ$^+$16] that studies PoIs classification based on their relevance, we are interested in PoIs for surrounding the contextual dimension. We then analyze the users activities such as dining, shopping, enjoying sports/exercises, tourism, and the like. So, we define five categories of activities as follows:

1. Regular activities: (Reg_Activity) that includes all the activities performed by a user with some regularity as to spend time at home, to work, or to go to Gym; activities that are performed in both MVP and OVP in the terminology of [PZG$^+$13],

2. Food activities: (Food_activity) this relates to the time spent by a user in food-related activities (pubs, restaurants)

3. Shopping activities:(Shop_Activity) this includes all the activities like habitual shopping, or other shopping related ones,

4. Exceptional activities:(Exc_activity)users occasionally go to hospital, bank, ect,

5. Tourism activities:(Tour_activity)this class is related to visiting leisure sites and tourism places.

Thus, we start by modeling the users activities related to the users clusters in order generate the user experiences.

Definition 16 *Let B be a set of activities, $B = \{a_1, a_2, a_3, ..., a_K\}$, each cluster r covers a number of activities that a users does in his/her stay locations. We can model the user experience, the individual's activities history, with a sequence of couples of cluster-activity:*

$$ExpU = (r_1, a_1) \rightarrow (r_1, a_2) \rightarrow ...(r_l, a_2) \rightarrow ... \rightarrow (r_l, a_q) \tag{6.2}$$

Where a sequence with $k = \sum_j |a_j|$ elements is denoted $k-$sequence and we indicate it also with $seq^{(k)}$.

A cluster r can characterize different user's activities $a_i \in B$, for instance:
$B = \{Work1, Gym1, Restaurant, Tourism, ...\}$, it means the clusters cover the PoI where a user has been at.
We use the couple (r_l, a_2) for differentiating activities especially when we find the same type of activity at different geographical position

In addition, an affiliation User-Cluster matrix M is built, where rows denote the users and columns denote the clusters as shown in Figure 6.8, where each entry M_{ij} represents the occurrence number of a user in a cluster. This matrix is used to construct the user's travel experience database $ExpU$ allowing us to extract relevant semantic between two users when applying the frequent patterns mining algorithm.

6.4.2 Frequent patterns mining

In this section, we analyze the frequent behaviour of users with similar interests whose are likely to conduct similar activities at similar places. So, we design a frequent patterns mining algorithm to find out the similar frequent k-sequences based on the level-wise algorithm and the data mining principles [AIS], Aggrawal and al[APY] [Zak] [HPYM]. In the first step, the algorithm reads the user's travel experiences $(ExpU_i, ExpU_j)$, after that it generates recursively the frequent $q-$sequences and checks their supports $(sup(e)$) which gives the number of times each sequence appears in the dataset intersection (Seq_{ij}) of two users, in that way we find all similar $k-$sequences shared between them. We have used feature vector to compare users activities in the user's travel experiences database. Two vectors F_i and $F\prime_j$ that are closely related will have a small distance and a large similarity. Even if this process is very expansive, we use it only once, thus it does not really affects performances.

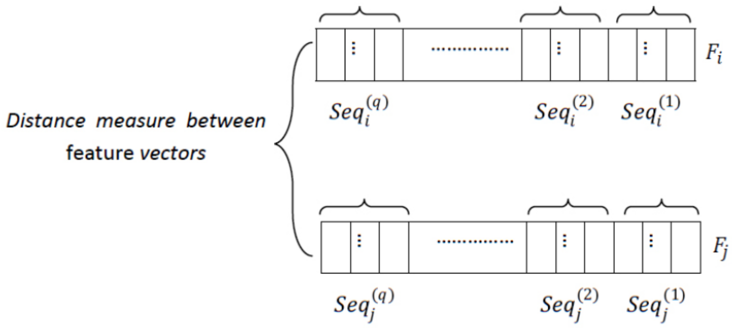


Figure 6.2: Illustration of the elements of Feature Vector

This algorithm can be expensive but here it scans once the dataset $(ExpU_i, ExpU_j)$ and then it finds the support of all k-sequences without need to scan again the database, due to this fact the algorithm is efficient.

```
Data: UExp_i, where i ∈ 1...n
Output: F, Seq_ij^(k) where i, j ∈ 1...n

foreach user u_i do
    foreach (UExp_j, where j! = i and j ∈ 1...n) do
        Seq_ij^(1) ← listofsequenceswheresup > min
        k ← 1
        repeat
            k ← k + 1
            Seq_ij^(k-1) determine all the set of candidate k−sequences Cand_k
            Seq_ij^(k) ← ∅
            foreach e ∈ Cand_k do
                Compute (sup(e))
                if sup(e) > min then
                    Insert(e, Seq_ij^(k))
                    Construct(F_i)
                    Construct(F′_j)
                end
            end
        until Seq_ij^(k)! = ∅;
        Find_Sequence_Max(Seq_ij^(k))
        Computedistance (d_ij)
    end
end
```

Algorithm 5: Finding out similar frequent k-sequences

6.4.3 Community discovery method

Communities are groups of nodes that probably share common properties and/or play similar roles within the graph [BLM+]. The communities may be correspond , for example, to groups of Web pages accessible over the Internet that have the same subject [FLGC], functional modules as cycles and pathways in metabolic networks [GA] ,[PDFV05], subdivisions in the food webs [Pim], [KFM+] and groups of people with some pattern of contacts or interactions between them [GN], [LN]. Here, we focus on modeling the mobile communities based on the users' interest to be able to accurately identify the network structure.

Definition 17 *let $G = (V, E)$ be a weighted graph describing the connection between mobile users, with V the set of nodes representing users and E the set of edges. An interest based mobile community, denoted $C_{M_i}(V_C, E_C)$, $\forall i \in 1..n$, is a subgraph of nodes with strong similar behaviour with respect to the spatial-temporel dimension, so that the adjacency matrix A of G is defined as*

$$A(i,j) = \begin{cases} w_{ij} & \textit{if user } i \textit{ and user } j \textit{ share similar interests} \\ 0 & \textit{otherwise.} \end{cases}$$

For our purpose, we compute the similarity matrix based on the dissimilarity index[Zho], the distance between two nodes given by equation 3 corresponds to differences between users' experience, then the distance matrix gives a perspective of the whole network taking any node as a viewpoint. Formally, the similarity measure is written as

$$w_{ij} = 1 - \frac{\sqrt{\sum_{k=!i,j}^{n} (d_{ik} - d_{jk})^2}}{(n-2)} \tag{6.3}$$

It follows that similarity measure will be large if i and j belong to the same community and small if they belong to different communities.

There are different measures to quantify the quality of a graph clustering [For]. Modularity [NG] is certainly the most widely used quality measure which can be described as the fraction of edges in a community, minus the expected value of the same quantity in a network with the same community divisions but random connections between the nodes. The modularity is based on assortative mixing measure [Newb] and it is defined as

$$Q = \frac{1}{2m} \sum_{i,j} (A_{i,j} - \frac{k_i k_j}{2m}) \delta(C_{M_i}, C_{M_j}) \tag{6.4}$$

where m denotes the total number of edges of the graph. The nodes are dived into communities such that node i belongs to community C_i. The function δ yields one if nodes i and j are in the same community, zero otherwise. We use modularity as a quality function and criterion to specify the best partition may be found.

We use an agglomerative algorithms to detect communities. In order to illustrate the relevance of our methodology, we apply three community detection methods:

6.4.3.1 Fast algorithm

defined in [LN] to identify community structure. Initially, the algorithm considers each node as a single community. Then it merges the two communities that maximize modularity and then updates the adjacency matrix A between this new community and all the other remaining communities that are connected to it. In our case, this merging process is continued until all the nodes are merged into interested based communities and modularity cannot be further maximized.

6.4.3.2 Random walk algorithm

Pons et al [PP06] have proposed an hierarchical agglomerative clustering method based on random walks. The walk-trap algorithm introduces the intuitive report that if a walker is in a community it has a strong probability to remaining in the same community at the following stage. have defined distance metric which is related to the spectral approaches which are based on the fact that two nodes belonging to the same community have similar components on the principal eigenvectors. The algorithm computes the connected components, and applies then an agglomerative algorithm which discovered communities separately on connected sub-graphs.

6.4.3.3 Community detection Algorithm based on Relevant Coefficient

We have proposed a community detection algorithm based on relevant coefficient that allows extracting knowledge about users activities [SMD15]. To this end, we define a relevance coefficient that computes the probability that two users share a same interest in the same place. thus, relevance coefficient reflects how much they participate ,together, several times at the same community.

We define a matrix L that This matrix expresses the behavior of users during his/her visits of different locations. So the matrix L_{ij} can be described as:

$$L_{i,j} = A_{i,j} * |sup(e)_{ij}| \tag{6.5}$$

Therefore, we define a factor that refers to the frequency of visit of user i and user j for each k-travel sequence:

$$f_{i,j} = \frac{L_{i,j}}{T_{sp \in (r_l, a_q)}} \tag{6.6}$$

Where T_{sp} ist he the total stay point visited given by $(ExpU_i \cup ExpU_j)$.

Our idea revolves around the basic principle of structure of community that a node has a high tendency to being members in community if it represents a meaningful or pertinent connection with a community. Now, we define a coefficient which we have called relevance coefficient:

$$\lambda(i,C) = \frac{\sum\limits_{j \in C} l_{i,j}}{\sum\limits_{j \in V} l_{i,j}} \tag{6.7}$$

$$= \sum_{j \in C} f_{i,j} \tag{6.8}$$

The detection of inter-community links is here based on the fact that such links are in less clustered areas. It is defined as the relevance value of node i in community c, divided by the total number of such possible values of relevance to all communities. Then, we rewrite the modularity in Eq. (6.9) as

$$Q = \frac{1}{2m} \sum_{c \in C} \sum_{i,j \in V} [L_{i,j} - \frac{k_i k_j}{2m}] \lambda(i,c) \lambda(j,c) \tag{6.9}$$

Such as

$$0 \leq \lambda_{(i,C)} \leq 1 \tag{6.10}$$

The nodes are divided into communities such that if node i belongs to only one community C, $\lambda_{(i,C)}$ is equal to 1; if node i does not belong to community C then $\lambda_{(i,C)} = 0$.

We exploit the relevance coefficient to reveal the community structure of a network using an agglomerative algorithm that we have defined in Algorithm 6.

Data: V
Initially $P^{(0)} = C_1^{(0)}, C_2^{(0)}, ..., C_n^{(0)}$
repeat
  Compute relevance coefficient between each pair of adjacent communities
  Select the two communities C_1 and C_2 of $P^{(k)}$ maximize modularity
  Create the new partition $P^{(k+1)}$
  Update S_{ij} for all adjacent communities to the new community.
until *have only one community*;
Select the best partition in communities of the resulting dendrogram which maximizes the quality function.

Algorithm 6: Relevance coefficient based method

6.5 Experimental results

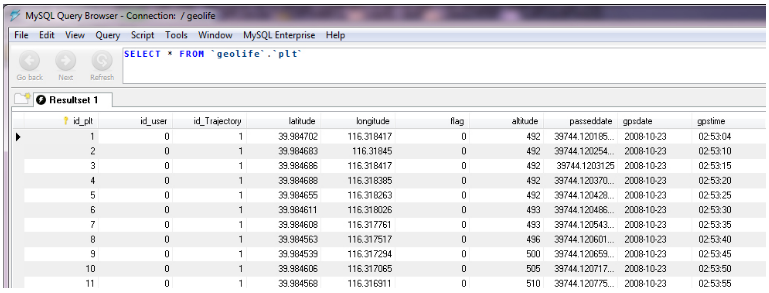
Finding a fundamental community structure is interesting in the sense that they can be useful for a better understanding of the behaviour of users. In this study, the experimental results allows to discover the community distribution and properties. We carried out experiments on the results obtained after applying pre-processing methods on GeoLife Dataset, as discussed below.

6.5.1 GeoLife overview

In this work we use a very large dataset collected in GeoLife project and released by Microsoft Research Asia [ZZXM] [Zhe]. The GeoLife dataset includes individuals movements in life routines, such as go to work or go to home, and also outdoor and entertainment activities. It contains trajectories taken with different GPS loggers and GPS phones in different sampling rates and contains latitude, longitude and height of every sample. It contains 182 users and a time period of five years from April 2007 till July 2012. In total, it contains 2153 trajectories. Majority of the trajectories were recorded with a sampling rate of 1-5 seconds or 5-10 meters. This dataset is wildly distributed in over 30 cities of China and even in some cities located in the USA and Europe, the majority of the data was created in Beijing, China[ZLC^{+}] [ZXM]. Table 1 summarizes the GeoLife characteristics. The available GeoLife dataset is a GPS log files which are a textual files. In this work, our preprocessing method has been tested on a final database that contains all explicit and implicit information that can be gained from the log files that represents the collected trajectories , where each row represents an individual GPS point, for each user we have an anonymized identifier, trajectories identifiers, latitude, longitude,and timestamped 6.3. Over this database we can perform our data analysis.

Table 6.1: Characteristics of mobility dataset

GPS dataset	Value
Time span	05 years
Subjects	182
Samples	24876978



id_plt	id_user	id_Trajectory	latitude	longitude	flag	altitude	passeddate	gpsdate	gpstime
1	0	1	39.984702	116.318417	0	492	39744.120185...	2008-10-23	02:53:04
2	0	1	39.984683	116.31845	0	492	39744.120254...	2008-10-23	02:53:10
3	0	1	39.984686	116.318417	0	492	39744.1203125	2008-10-23	02:53:15
4	0	1	39.984688	116.318385	0	492	39744.120370...	2008-10-23	02:53:20
5	0	1	39.984655	116.318263	0	492	39744.120428...	2008-10-23	02:53:25
6	0	1	39.984611	116.318026	0	493	39744.120486...	2008-10-23	02:53:30
7	0	1	39.984608	116.317761	0	493	39744.120543...	2008-10-23	02:53:35
8	0	1	39.984563	116.317517	0	496	39744.120601...	2008-10-23	02:53:40
9	0	1	39.984539	116.317294	0	500	39744.120659...	2008-10-23	02:53:45
10	0	1	39.984606	116.317065	0	505	39744.120717...	2008-10-23	02:53:50
11	0	1	39.984568	116.316911	0	510	39744.120775...	2008-10-23	02:53:55

Figure 6.3: Mobiles Paths Database

6.5.2 Interest Based Mobile Communities Extraction

The stay point detection algorithm extracts the users whose position is tracked for one year from 01-January-2009 to 31-December-2009, resulting in 70 users for which we have 23060 stay points. Figure 6.4 shows some GPS points of user having $id = 0$ and illustrates the extracted stay points. It is clear that the algorithm omit some points like road crossing that user regularly passed during his/ her daily movement. After that, we have characterized users mobility according to their radius of gyration. Figure 6.5 presents the results of the radius of gyration evaluation over the total duration of the traces, it shows that, in geolife, users travel over limited neighbourhood, a few users regularly cover hundreds of kilometers.

Pt	User	Traj.	Latitude	Logitude	Time
1	0	1	39.984702	116.318417	2008-10-23 02:53:04
2	0	1	39.984683	116.31845	2008-10-23 02:53:10
3	0	1	39.984686	116.318417	2008-10-23 02:53:15
4	0	1	39.984688	116.318385	2008-10-23 02:53:20
5	0	1	39.984655	116.318263	2008-10-23 02:53:25
6	0	1	39.984611	116.318026	2008-10-23 02:53:30
7	0	1	39.984608	116.317761	2008-10-23 02:53:35
8	0	1	39.984563	116.317517	2008-10-23 02:53:40
9	0	1	39.984539	116.317294	2008-10-23 02:53:45

Figure 6.4: GPS trajectory and the extracted stay points

- Users with $r_g <= 35km$. They might be identified as urban mobile people as the diameter of the Beijing periurban area is very approximately around $35km$.
- Users with $50km <= r_g <= 150km$. Some users have travelled over $50km$ and/or outside the suburban area of the city of Beijing.

For our purpose, we will select only the users whose radius gyration are less than or equal to $150km$, resulting in 42 users for which we have 10960 stay points.

Then, we apply the identification of PoIs process, described in Section 4.1. Figure 6.6 and Figure 6.7 show a cumulative characterization of the activities in PoIs identified for two sampled users and we plot their points of interest on the map. We notice the users tend to engage in several type of activities, they practice regular activities (Reg_Activity) such as going home and work, enjoying sport and gym, we also remark a several food activities (Food_activity)

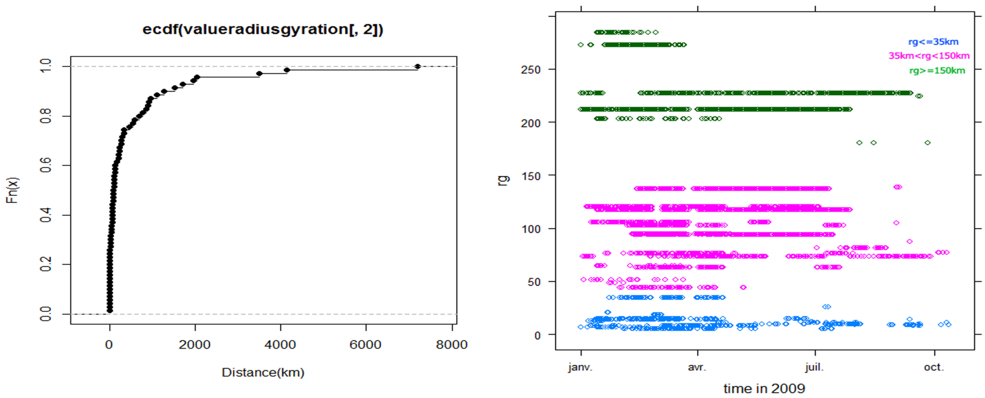


Figure 6.5: a)Empirical Cumulative Distribution of radius gyration of the extracted 70 users b) The distribution of radius of gyration rg(t) for users during one year

such as dining in a restaurant, enjoying time in pub and caffé. Concerning shopping activities (Shopp_Activity), users frequent use a specific commercial area or shopping places. The exceptional activities (Exc_activity) illustrate that users occasionally go to hospital, bank, etc... At last, users (Tour_activity) travel far distances to visit cultural places, public attractions and most interesting tourism places in Beijing like Summer palace, Bird's Nest, Olympic park, and so on, but such travels are occasional. In the example of the user 167 , we notice the absence of shopping activities due to the fact of missing data in GPS trajectories. To deal with this problem, we have generated the missing data using the algorithm proposed in [BE], although it adds some inaccuracy.

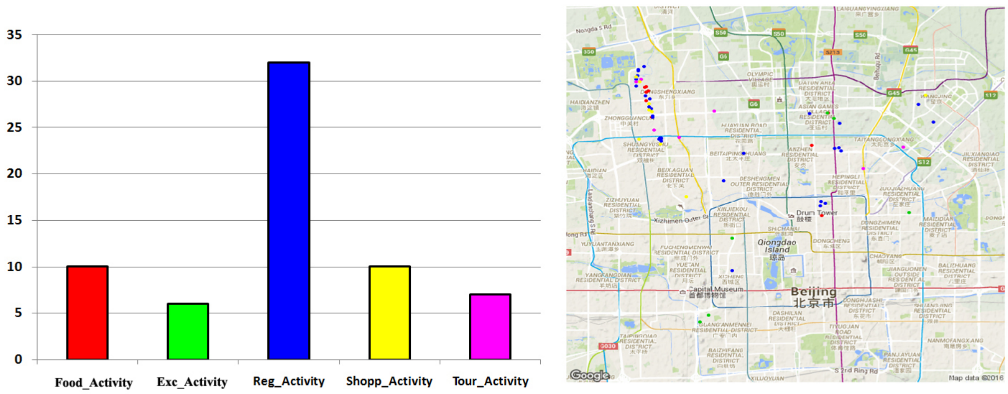


Figure 6.6: a)Profiling of user 126 for different activities b)Points of interest for user 126

The semantic meaning of the obtained clusters reveals what a user can do at his/her PoIs as shown in figure 6.8. In this dataset, we also observe that users share not only the interests, but also some visited points. This happens, for example, for the most famous tourism location in Beijing. Furthermore, in order to compare the frequent users activities behaviour contained in the sequence pattern, we maintain the set of activities that a user can does at each cluster. In this way, the detection of groups that share a common interest with respect to the special dimension can be overcome.

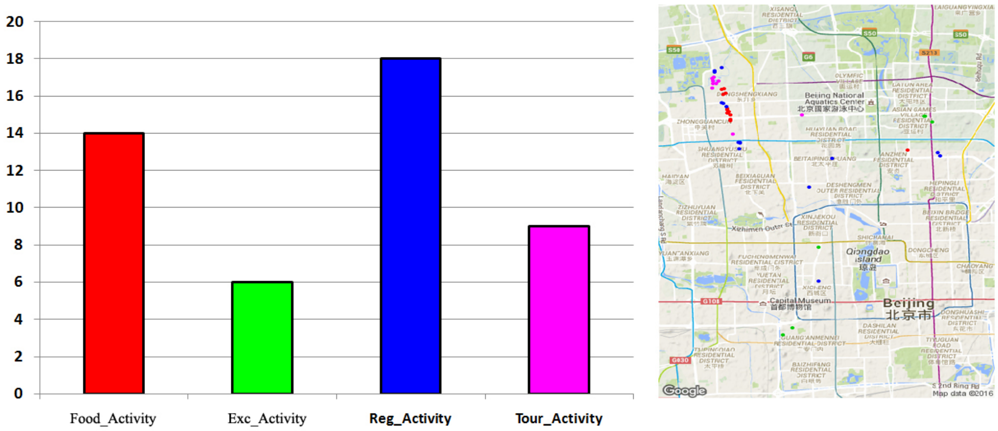


Figure 6.7: a)Profiling of user 167 for different activities b)Points of interest for user 167

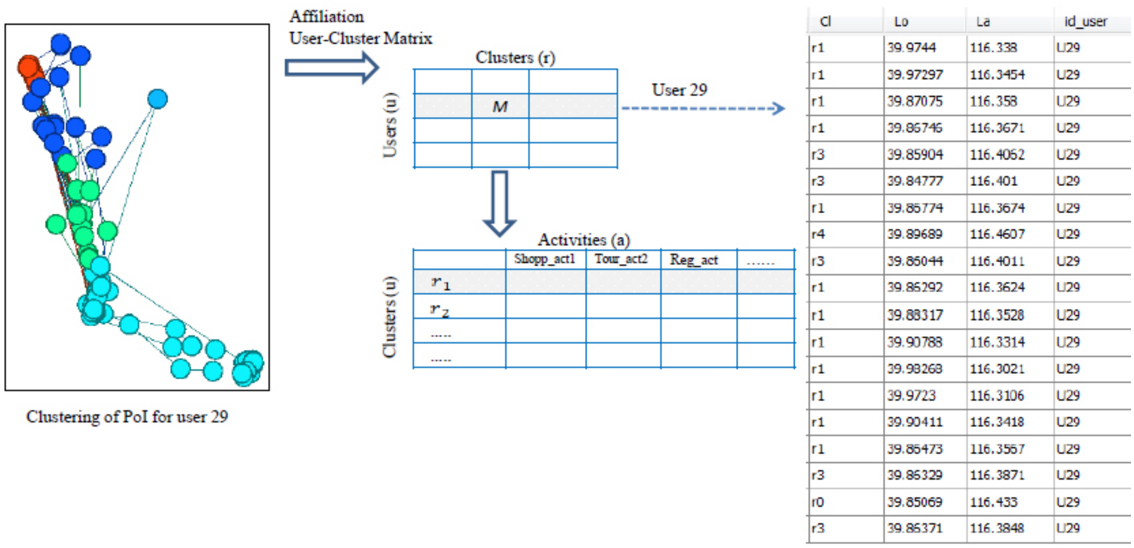


Figure 6.8: An example about extracted activities does user practice within different clusters

Discovering frequent sequence patterns allows extracting knowledge about user's activities. As example, in figure 6.9,we illustrate how algorithm 2 identifies the same common pattern between user i and user j by counting the number of occurrences registered as support, the next step is to permute and find all the k-sequence using Find_Sequence_Max($Seq_{ij}^{(k)}$) which can localize support counting and reduce the search space, and in each iteration we obtain the F_i and $F\prime_i$ elements, and thus the algorithm quantifies the distance between users, for example, we have find that user 126 is more similar to user 167 ($w = 0.43$) than user 29 ($w = 0.21$) as user 126 shares more travel sequence with user 167. The extracted features are fundamental for any prediction activity, either at user level or at community level, as discussed in the conclusions.

In pattern discovery step, we have identified community structure and analysis the users with similar interests. As a result, the extracted graph is a weighted and undirected graph in

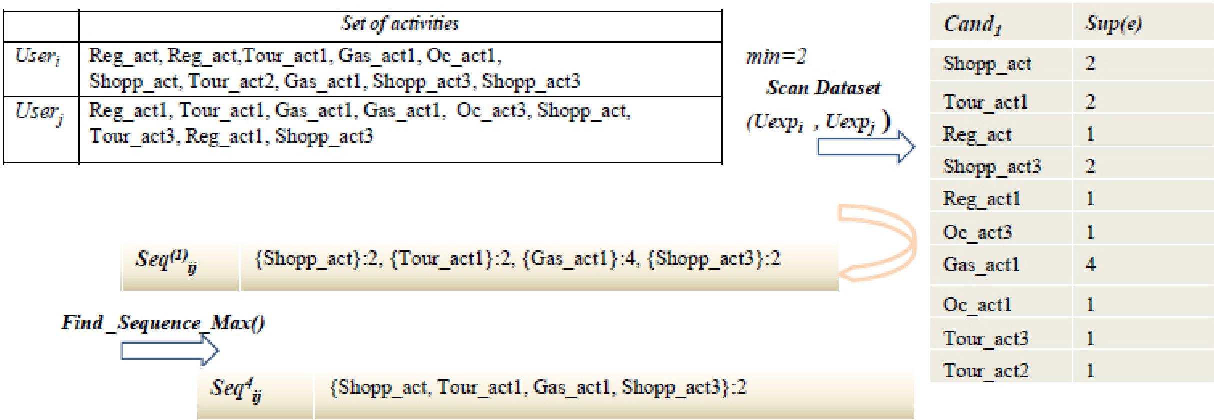


Figure 6.9: Frequent mining strategy to find similar users experiences.

which there are 42 nodes and 174 edges (See figure 6.10). Results shown in table 6.2 compare the ratio between the sum of weights of intra-links (links connecting nodes that are placed in the same community) and the sum of weights of inter-links (links connecting nodes from different communities), when applying fast algotithm, in order to illustrate if communities exhibits Radicchi strong or weak community property [RCC$^+$].
The quality of partition of the three community detection algorithms illustrated in table 6.3. The algorithm based on relevance coefficient finds 06 communities, with a maximum value of modularity of 0.45. Fast algorithm discovers 06 communities and the modularity equals to 0.33. The random walk algorithm finds 05 community structure with a value of Q_{max}=0.28 . All the methods detects a significant community structure. The method based on relevance coefficient shows a significantly stronger increase in modularity than expected if we use fast algorithm due to the fact that relevance coefficient provides more accuracy about the presence of nodes with high frequency in a given community.

In what follows, we describe the discovered communities and their significance:

- *The algorithm finds dense communities formed with almost every node strongly linked to every node else: the interest based mobile communities.* Figure 6.10 shows two interest based mobile communities (community 5 (blue nodes) and community 1 (red nodes)), where users are prospective tourists because they have visited some famous location in Beijing. In our analysis, users have a strong correlation due to the similarity in their tourism-related activities, for instance, in community 5 a user may do tourism activities such as *tourism* $\rightarrow$ *shopping* $\rightarrow$ *cultural activity* $\rightarrow$ *commercial area*. Therefore we observe the creation of communities sharing a common interest on tourist attraction and public locations (Summer Palace, Bird Nest, Plazma center, Olympic park, ect) that have been visited by some of them. We can make the same observations in community 1 where the sequence of activities has the similar semantic but it is different regarding the locations. Note that the average distances between the visited locations covered by the members of these communities is 38km, which explains their nature in terms of interests.

- *Communities in which nodes exhibit high physical closeness*, because they express the existence of individuals who have practised similar activities in the same locations during the studied period. Communities 6 and 2 (the nodes labeled by clear blue represent the second community, and those labeled by yellow represent the sixth community) express the closeness of participants or physical closeness which eventually results in better interaction between them. In this kind of relations, users visited nearby locations, for example, node 22 and node 26, in community 6 are closely related to each other due to the fact that their PoIs are located in the same places which are "Tsinghua University" and the nearby restaurants.

- *Communities are of various nature as some nodes are bit close physically to each other and they present an external contacts with others communities.* Indeed, if we take the example of community 3 (green nodes are belonging to this community) and community 4 (it contains 7 mauve nodes), in which users tend to react with similar behaviour since they are found in the same locations of work as well as same common tourist attraction.

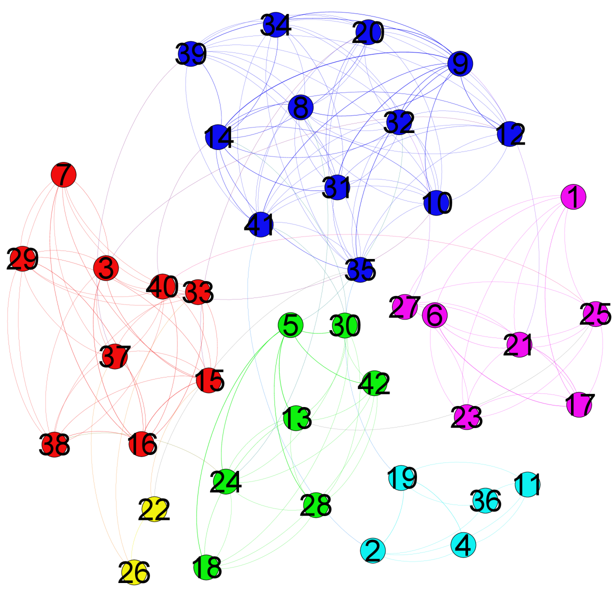


Figure 6.10: The results of detecting communities using Fast algorithm

So, as demonstrated with the GeoLife dataset, the proposed methodology allows discovering mobile communities according to their location histories and their preferences or (activ-

Table 6.2: Comparison of inner community edges and the inter community edges for the network partition.

Communities	Number of users	Inner	Outer
Community 1	9	10196 (93.60%)	696 (6.39%)
Community 2	5	1546 (92.02%)	134 (7.97%)
Community 3	7	9392 (97.23%)	267 (2.76%)
Community 4	7	4824 (94.66%)	272 (5.33%)
Community 5	12	25640 (97.39%)	685 (2.60%)
Community 6	2	432 (61.89%)	266 (38.10%)

Table 6.3: Qaulity of partition by applying Relevance coefficient algorithm, Fast algorithm, and Random walk algorithm.

Community Method	Communities number	Q Function
Relevance coefficient	06	0.45
Fast algorithm	06	0.33
Random walk	05	0.28

ities), and to classify each user within a community according to her/his sequence of activities.

In Community 2, we can notice that users who have a very high physical proximity have been at a common locations for regular activities and food activities. For that reason, the daily movements of individuals are most often characterized by small distances. In Community 3, we observe that members are users with similar interests and some common locations. We can thus think that they may know each other and know the habits of other community members (colleague, friends, family).

6.6 Conclusion

In summary, our work aims to identify interest based mobile communities structure for mobile users. We have proposed a mobility data mining methodology. Our methodology consists of three main steps. First, we detect stay points from individuals' traces and we characterize users' movement according to their radius of gyration measurement. In the second step, we extract and categorize points of interest into several meaningful classes, using density-based clustering algorithm and incorporating the semantic of users' activities. We then present a frequent mining algorithm to compare between users' travel experiences. In addition, we compute the similarity measurement and employ a detection community method to identify the community structure propriety of the network.

We demonstrated the validity of the proposed methodology to extract interest based mobile communities in a real case. We were able to derive a meaningful description of existing mobile communities, and to represent a different semantic relationships associated with users in the network.

Indeed, the extracted features can be very beneficial for prediction. In particular, by extracting the sequence of activities of a user and the share of interests with some other users, we can predict the likelihood the user behave in a particular way and define the probability to choose a nearest location to his group of interest. This can provide a good basis for applications as recommendation or network management in complex networks of mobile users.

Chapter 7

A Novel Prediction Method for Analyzing Users Future Movement

7.1 Introduction

Predict individual's next movements using his/her past history and also the history of people related to him/her is one of the most interesting research areas in computational social science. Wang et al [WPS+11] have studied and analyzed the trajectories and communication records of 6 Million mobile phone users. The authors have proved, by combining the measurements of network proximity and mobile homophily, that the similarity between two individual's movements is strongly correlated with their proximity in the social network. In [PZ15], Pang et al have determined the check-in geographic regions and identified communities of user's friend on the tweeter network, and have demonstrated that communities' influences on users' mobility are stronger than their friends' and each user is only influenced by a small number of his/her communities. Kamini and al [KMG16] proposed a new prediction algorithm based on users interest profile and the mobility history of the community. They have illustrated that a single user in his/her own visiting behaviour tends to be more conservative than looking at himself whithin a croud of people and the overall community tends to deviate from its regularity more easily than a single user.

In our previous work [DBSG17], we have identified Interest Based Mobile Communities, called IMoComm, for mobile users. Interest that seems to be the main reason that motivates individuals to move from one place to another. In fact, the extraction of a user's sequence of activities and the share of interests with some other users allows to predict the likelihood that the user will behave in a particular way and to define the probability of choosing a location close to his/her group of interest. In this work, we aim to improve such prediction by exploiting additional available information included in the IMoComm. Intuitively, an individual tends to join a community of his/her interests that is varying over time but his/her move is strongly connected to his/her social preferences, career goals, and daily life habits. Thereby, the extrac-

tion of community link patterns helps predicting his/her future movement by incorporating useful information conveyed by users communities ties while tracking his/her mobility history. The link prediction problem has attracted immense interest in recent years, and a variety of techniques that operate on the graph/hypergraph structure of social networks are proposed. For a full review of the state of the art in link prediction in social networks, see Peng et al[WXWZ15]. In our contribution [[dr17], we deal with such link prediction issue: we analyze the dynamics of Interest Based Mobile Communities and we build our prediction model for user's future movement by exploiting the abstraction level of users correlation patterns and their IMoComm.
The following of this chapter is structured as follows. Section 2 states the problem. Section 3 presents the preprocessing of data set used in our work, and general statistics. In section 4, we introduce the prediction model based on community related features. In section 5 we discuss the results of experiments made on the available individual trajectories. Finally, conclusions are given in Section 6.

7.2 Problem statement

In daily life, people participate to various communities (colleagues, family, friends, ect). Their mobility is driven by their interest and need to practice different activities with other people depending on the type of the community they share (colleagues, friends, food, shopping, tourism, sport, ect). Hence, we study the human mobility behaviors from the perspective of network science, in particular the goal of this contribution is to study how to use the knowledge gained from the IMoComm membership of each person and how it can be used to predict the community evolution (future links). Firstly, we perform an unsupervised task to extract the link pattern between people that distinguishes meaningful Interest Based Mobile Community structures and expresses the individual mobility behavior while sharing a common interest regularly or from time to time. We study then the link prediction problem using the resulting learning graphs, and we formulate our problem as follows: we start from a weighted graph $G(V, E)$ where V and E are sets of nodes and links resulting from the IMoComm, respectively. Let the subgraph $G[t, t\prime]$ denotes a snapshot of the social network between two times t and $t\prime$, such $t < t\prime$. We then predict the likelihood of a future connections between nodes and links in the network $G[t_1, t\prime_1]$. In other words, the link prediction aims to infer which new mobile community a user is likely to be at in the near future. So, predicting prospective links or deleted links in IMoComm graph for a future period is fundamental. Thus, we develop an approach to link prediction based on the analysis of community related features in the human mobility context.

A graphical representation is given in figure 7.1, in which solid links indicate that a user was already member of an IMoComm during the period $[t_0, t\prime_0]$, and dashed lines are used to indicate links that might appear during the interval $[t_1, t\prime_1]$ when users will move toward different communities.

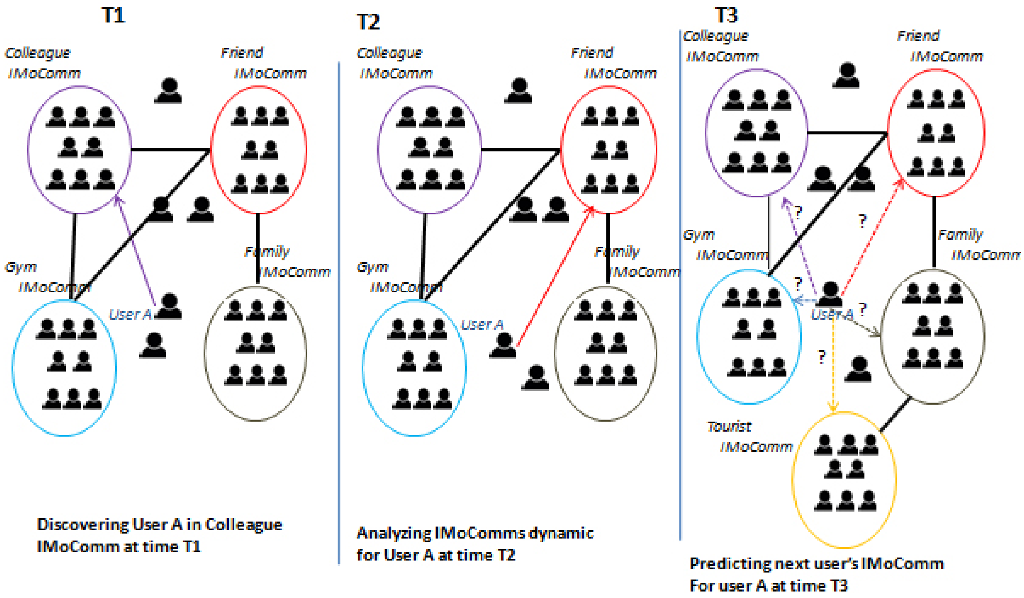


Figure 7.1: An example to explain the link prediction problem in Interest Based Mobile Communities Graph

7.3 Data set preprocessing

Human trajectories systems make use of location extraction techniques from geospatial data to identify locations that have meaning and importance to the users. Here, we have implemented stay points extraction method [LZX+08] in order to extract meaningful stay of individual who has spent a considerable time on a geographic region, for more details see [DBSG17]. The algorithm results in 23060 stay points for all users whose position is tracked in 2009. Figure 7.2 shows the total number of stay points per week in 2009 and illustrates the number of users having an accurately users' tracked position during 2009, we remark the lack of some users traces, although we have generated missing data using the algorithm proposed in [BE10]. Besides, we limit our analysis to GPS data collected in the region around Beijing. However, the rate at which users provide a new location point is not constant and not all users are present on all days (see Figure 7.4). It is therefore reasonable to select the active periods for our experiments.

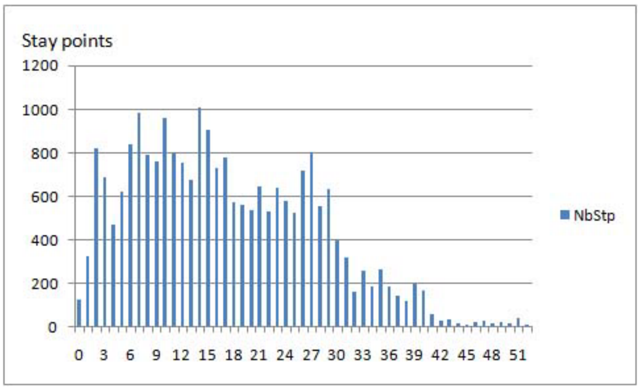


Figure 7.2: Number of stay points per week in 2009.

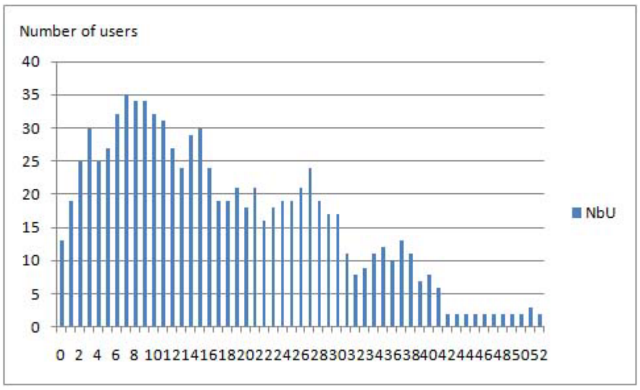


Figure 7.3: Number of users per week in 2009

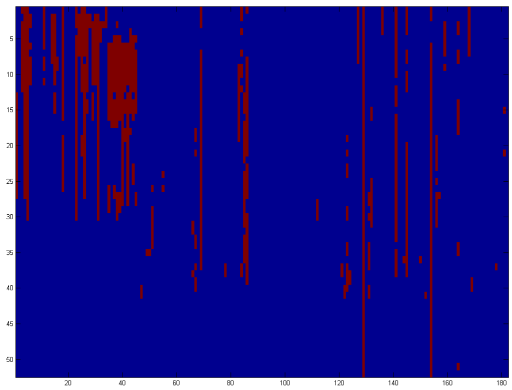


Figure 7.4: Users movments during 52 weeks

7.4 Location Prediction based on mobile communities

The goal of our model is to predict the next IMoComm with which the user is going to interact at a certain day of the week, exploiting the learned visiting behavior of the user, his/her daily activities, and his/her relation between some users that share a common interest.

7.4.1 User's communities pattern extraction

In order to achieve our ultimate goal, we start by discovering a learning graph that will be used to predict future potential links. Firstly, we apply DBScan algorithm [EKS+96] on stay points, the clustering parameters have been defined empirically ($minPoints = 3$, $\epsilon = 0.02$ dd), and we add semantics of location which imply the activities being carried out in each cluster in order to understand the relationship between the geographic location and the users activities. Basically, the individual's activities history consists of a sequence of couples of cluster-activity, thus

$$ExpU = (r_1, a_1) \rightarrow (r_1, a_2) \rightarrow ...(r_l, a_2) \rightarrow ... \rightarrow (r_l, a_q) \tag{7.1}$$

where r_j is the j^{th} cluster covering a number of activities $a_i \in B$,
$B = \{a_1, a_2, a_3, ..., a_K\}$, that a users does in his/her stay locations. We have shown [DBSG17] that interesting locations for people can be grouped in several categories (regular activities, food activities, exceptional activities, Shopping activities, and Tourism activities). This suggests that people share common PoIs (points of interest where a user has been at), but even more they share common interests. We thus mine the frequent behaviour of users with similar interests [DBSG17] and compute the similarity of two users in terms of similar activities at similar places using Ecludien distance between $UExp_i$ and $UExp_j$. From figure 7.5 a) we see that user of $id = 0$ and user $id = 30$ had revolved around similar PoIs in physical places, during 06/04/2009 for a timestamp from $08:00:00am$ to $12:00:00pm$, which result in social ties weighted with similarity value equal to 0.49 and these users have formed one IMoComm, especially in the specific timeline (timeline of work, or timeline of different daily activities). Figure 7.5 b) shows correlation between users who visits the same tourist places in weekend (many users have been around the very famous park of summer palace in same time during the weekend 18/04/2009). The idea here is that users sharing more common interests with a specific user should be more similar to him/her than users with less common interest.
Furthermore, we find an appropriate set of time intervals for the set of users since individuals are most likely to belong at group in a given time step, and apply community detection algorithm [LN] on each snapshot and matching communities applying the algorithm described in [MLH02], therefore we find a fundamental community structure and extracted features about development of human mobility and their IMoComm over time, we will discuss this finding in section 7.5.1.

In this phase, we have created the learning graphs $G_i = (V_i, E_i)$, $\forall i \in 1..n$, describing the connection between mobile users, with V_i the set of nodes representing users and E_i the set of edges, which distinguishes meaningful Interest Based Mobile Community structure $C_{M_j}(V_{M_i}, E_{M_i})$, $\forall j \in 1..f$.

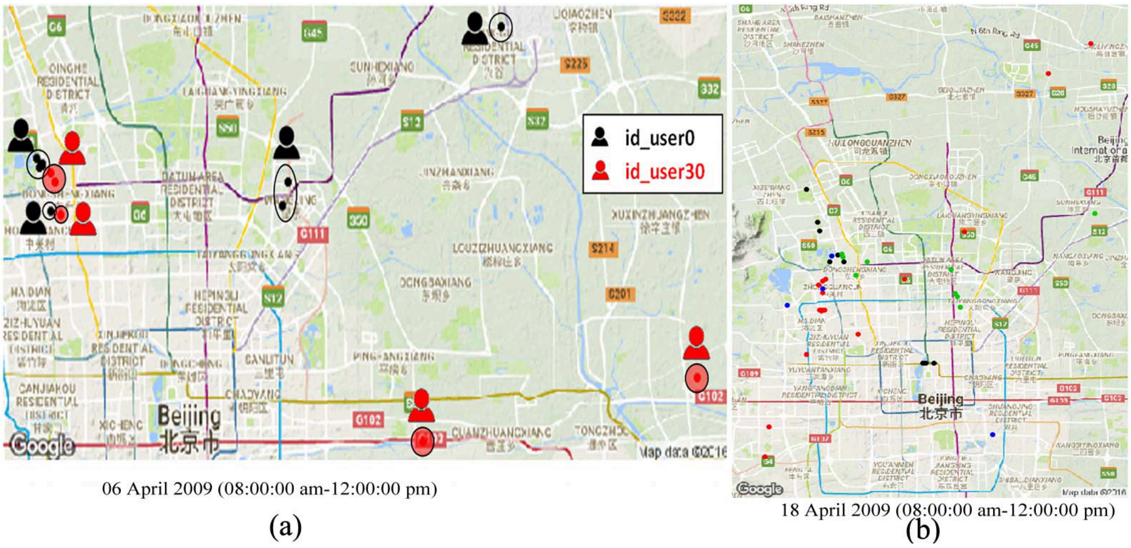


Figure 7.5: Illustration of users similarities

7.4.2 Community related features

The community information provided by the detected IMoComm provide powerful features for predicting individual's daily behaviors which are largely dependent on his/her preferences, his/her activities and his/her social relations. As a matter of fact, in our work [DBSS16], we are able to consider community features, which have interesting impact on the user's mobility:

- two users are regularly linked (strongly related) within a community are more likely to visit, in near future, the same IMoComm than two users who have no contact and/or don't share similar interest in similar place (weakly connected).
- Besides in communities that they attend regularly (such as communities of colleagues, friends, family, etc.), users exhibit a large similarities with the members of other communities they belong to, such communities are clearly more likely to affect more the behaviour than a community visited occasionally.
- Some groups have weakly connected links, thus their members are very varying over time (different community members in each snapshot).
- The more the same individuals form a community several times in specific period, the more regular and social the community is considered to be.

Our model accommodates these aspects as discussed in the following sections.

7.4.3 Prediction Model

Many methods for link prediction based on structural similarity between nodes have been proposed since similar nodes are likely to have neighbors in common and they are more likely to have the same relations in the near future[LNK07]. Therefore, in our work, we have used a topological measure for weighted graph to calculate the likelihood score of any pair of nodes u and v

$$score^{Weighted}_{(u,v)} = \frac{\sum_{w \in \Gamma(u) \cap \Gamma(v)} \sqrt{A_{uw} * A_{vw}}}{\sum_{w \in \Gamma(u)} A_{uw} + \sum_{w \in \Gamma(v)} A_{vw}} \tag{7.2}$$

Where $\Gamma(u)$ is the set of direct neighbors of node u in $G[t_0, t\prime_0]$, $\Gamma(u) \cap \Gamma(v)$ is the set of common neighbors of two nodes u and v in $G[t_0, t\prime_0]$.

The prediction function P that indicates likelihood of nodes (u, v) being in E_{new} can be used for ranking all possible edges according to their probability.

$$P_{(u,v)} = \frac{\sum_{i=1}^{nbrday} score^{Weighted}_{(u,v)}}{nbrday} \tag{7.3}$$

The number of days $nbrday$ is set according to the selected training intervals.

Our prediction method is based on the assumption that human mobility is affected not only by person's travel experience but also by his/her movement towards Interest-Based Mobile Communities. Initially, we divided the extracted pattern in two parts, $G_{leraning}$ and G_{test}, respectively, and select a learning period. For a sequence of snapshot; $< G_1[t_0, t\prime_0], G_2[t_1, t\prime_1], ..., G_l[t_l, t\prime_l] >$, in a given learning period, we compute the *probability list* for each missing links or links to occur in future. The algorithm computes the likelihood of nodes for each temporal graph and then generates the whole graph applying an aggregation steps. The use of a graph aggregation produces the overall structure of the underlying graph during the learning period and captures semantic knowledge not only about individual nodes and their connections but also about groups of related nodes. Thus, we can recognize the times a node has appeared in a community over time in order to make a decision about its community type. For example, given two nodes u and v that belong to the same regular community reg_1, their link (u, v) has a strong chance to appear in next time. If these two nodes belong to different regular communities reg_1, reg_2 the link might be formed in future time. Finally, whereas if these two nodes belong to the same or two different occasional communities oc_1 or

oc_1, oc_2 respectively, they do not have a strong priority, and the link between them is most likely won't occur in next period. Using community attributes helps in predicting the IMoComm that will may be visited during his/her next move. Finally, the algorithm takes the global probability list and sort it in decreasing order of the likelihood $P(u,v)$ and of the community types features. So, the k links in the top are most likely to exist.

Data: $< G_1[t_0, t\prime_0], G_2[t_1, t\prime_1], G_3[t_2, t\prime_2], ..., G_l[t_l, t\prime_l] >$, $id_{user} \in U$

Output: *PredictList*, L

Select the learning graphs :$G_1[t_0, t\prime_0]$,$G_2[t_1, t\prime_1], .., G_{l-1}[t_m, t\prime_m]$

foreach $G_i[t_i, t\prime_i]$ **do**

  $A \leftarrow A_{G_i}$

  compute $score_{u,v}^{Weighted}$

  read (H) /* History of user's communities

aggregate($G[t_0, t\prime_0]$,$G[t_1, t\prime_1], .., G_{l-1}[t_m, t\prime_m]$)

compute($P_{(u,v)_{agg}}$)

$CommunityTypes(u) \leftarrow$ generate(assign($CommunityTypes, id_u$))

$CommunityTypes(v) \leftarrow$ generate(assign($CommunityTypes, id_v$))

compute E_{wrong}, $E_{positive}$ in $G_{l-1}[t_l, t_l\prime]$

PredictList $\leftarrow$ Insert ($P(u,v)$, id_u,id_v, $CommunityTypes(u)$,$CommunityTypes(v)$)

Sort *PredictList* in descending order of the likelihood $P(u,v)$ and of the community types

$L \leftarrow$ Get top k links in *PredictList*

Validation using $G_l[t_l, t\prime_l]$ for test

Algorithm 7: Link-Prediction Algorithm

7.5 Experiments

7.5.1 Communities and mobility

In order to study and predict the dynamics of individuals and investigate their communities evolution in human mobility domain, it is essential at first to identify the evolution characteristics of this complex network in particular the occurrence of new links and duration of interaction of their entities. For example, citation networks have a small number of evolution steps that is a snapshot per year, while the biological networks have more and specific details of evolution. To this end, we have made several empirical tests to distinguish the dynamic features and the social aspects related to the evolution of IMoComm. To the best of our knowledge, this is the first work ,in human mobility area, that analyses users mobile communities and their dynamics:

1. The time step for each snapshot have to be taken for different timing that are closely related to the nature of the individuals interactions and their daily activities (according to the time-line of works, food, meeting friends, ect).

2. Relevant communities are created from the aggregation of all the links that appear and reappear at least twice during successive week days,for different timing, for all the studied period. This link type corresponds to regular human interactions, such as interactions between colleagues in work. This aspect is present in the blue and the red communities illustrated in Figure '7.6.

3. If two individuals perform the same activity in the same place only once, we consider that they are weakly connected and their link is not presented in the aggregation graph. However, we can take into account such links if they belong to public group and if they will help to characterize social aspect of human mobility. Moreover, this link type belongs sometimes to occasional IMoComm that exhibit a dense local structure around public places (such as tourist and cultural places). As evidence of such property, we have extracted a dense subgraphs during some weekend days; we have $Q = 0.393$ during 12/04/2009, $Q = 0.534$ during 19/04/2009, and $Q = 0.33$ in 26/04/2009.

4. The detection of IMoComm from the aggregate graph increases modularity and allows to identify a set of relevant communities.

5. The detected disjoint subgraphs in steps of daily time permit to discover overlapped communities on the aggregate graph. This is due to the fact that, in daily life, individuals can belong to multiple IMoComm but their number remains limited (daily communities). Figure 7.6 illustrates the disjoint groups for which we have selected few users who have continuous data collected for at least three consecutive days. We limited our study to the users that have continuous data collected during the selected period and we filter isolated results (see movement of users in Figure 7.4).

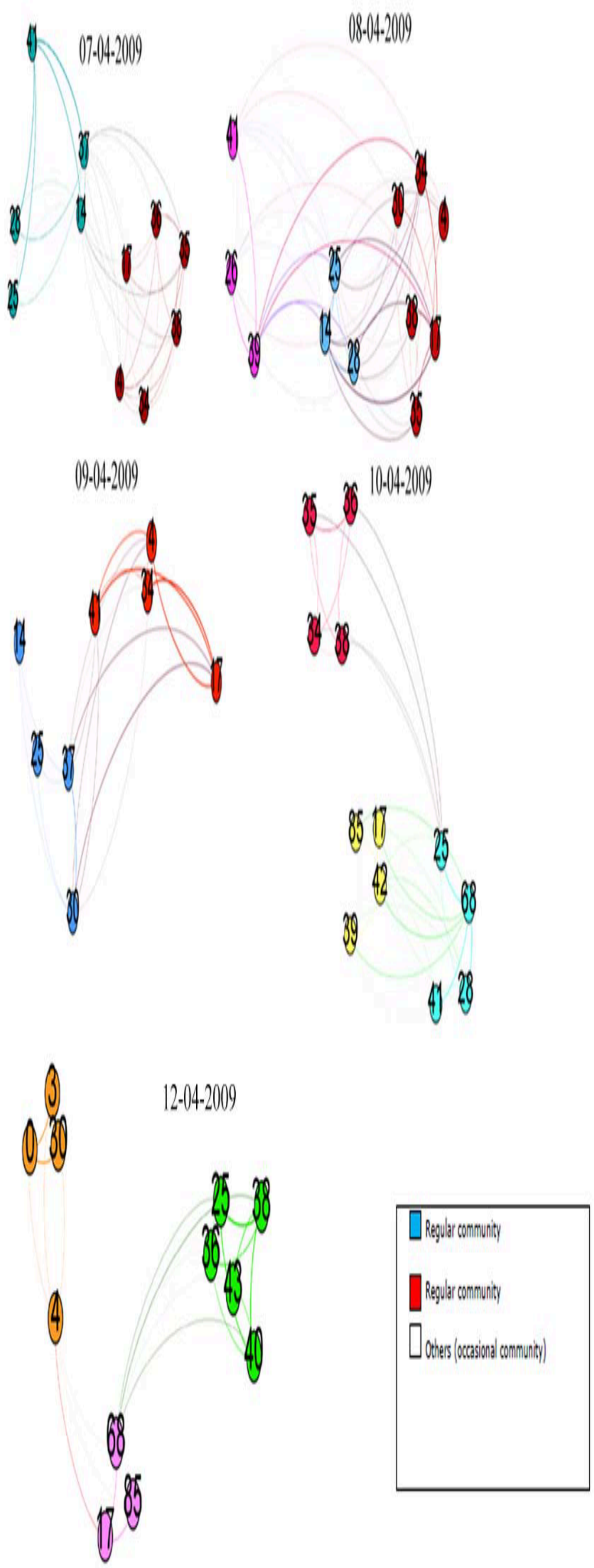


Figure 7.6: Dynamic of IMoComm over four days (from 07/04/2009 to 10/04/2009) and a day of a week, the time period is from 08:00:00 am to 12:00:00 pm

Table 7.1: Training and test periods for link prediction for three datasets

Datasets	Phase	Period	Edges	Nodes	Comm	Temporal sequence of graphs
Dataset1	Training phase	From: 01/04/2009 to: 10/04/2009	346	182	03	$G_1[01/04, 03/04]$, $G_2[06/04, 10/04]$, $G_3[13/04, 17/04]$
	Testing phase	From: 13/04/2009 to: 17/04/2009	212	182	03	
Dataset2	Training phase	From: 06/04/2009 to: 18/04/2009	420	182	04	$G_1[06/04, 10/04]$, $G_2[13/04, 18/04]$, $G_3[20/04, 25/04]$
	Testing phase	From: 20/04/2009 to: 25/04/2009	246	182	04	
Dataset3	Training phase	From: 10/04/2009 to: 25/04/2009	598	182	04	$G_1[10/04, 19/04]$, $G_2[20/04, 25/04]$, $G_3[26/04, 30/04]$
	Testing phase	From: 26/04/2009 to: 30/04/2009	112	182	03	

7.5.2 Prediction results

To predict a link, we select a training period and first extract the topological and community features for the temporal graphs, and then build the prediction model. Hence, given the selected temporal graphs $G_1[06/04, 10/04]$, $G_2[13/04, 18/04/]$, $G_3[20/04, 25/04]$, we partition them in training and test sets. The choice of intervals has been made in an empirical way. We denote the training interval to be 11 days:[06/04,10/04] and [13/04,18/04/]. We take $G_2[13/04, 18/04/]$ for labeling and we check that each pair (u, v) either represents a positive example (link exists) or a negative example (link does not exist). Thus the test graph $G_3[20/04, 25/04]$ is used to validate if a link exist or not(see Table 7.1).
From our dataset, we combines three datasets with different characteristics: *Dataset1* considers only working days, *datset2* includes a weekend day (Saturday), while *dataset3* includes four weekend days.

At the community level, our approach allow us to recognize the expected user's communities at the next step and to understand how a user plans his/her next move from his/her IMoComm's perspective. As we can see from Figure 7.7, where we have used *dataset1*, mobility history of user's communities extracted with our approach indicates that, despite the diversity of their travel history, humans follow, in most of case, simple reproducible pattern and have small number of communities. Thus the prediction process, which recognizes the most frequent communities of individual's travels, can predict movement of users and allows to characterize the common mobility behavior within his/her groups in the near future,(see Figure 7.7 b)). Figure 7.8 a) shows the type of predicted links between users using the IMoComm based approach. For instance, the probability that user $id_{user} = 38$ will join his/her regular community ($Reg_{IMoComm2}$) is equal to 0.446, and he/she may also move to $Reg_{IMoComm3}$ at the next step with probability 0.369, while probability to move to $Oc_{IMoComm5}$ is 0.163 and therefore it is not selected in the predicted list. This confirms that an individual move usually to some of his/her regular IMoComm, while it is unlikely that he/she will go to some occasional communities.

Thus, the proposed approach is very useful for predicting user's mobile behavior with regard to his/her next IMoComm. However, the algorithm needs more users attributes to be able to recognize the formation of new groups in near future. We mention also the problem of matching communities in dynamic complex networks which is an NP-Hard problem that we don't study it here.

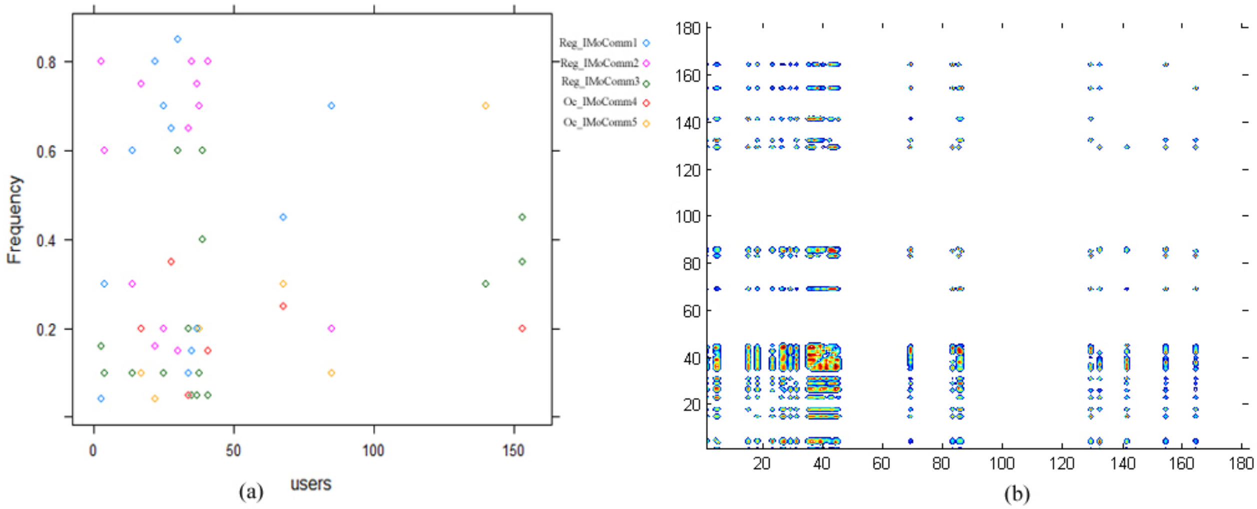


Figure 7.7: a) Distribution of mobility history of user's communities extracted when analyzing the dynamic of IMoComm b) Prediction of users' future link and thier excpecting IMoComm from 13/04/2009 to 17/04/2009)

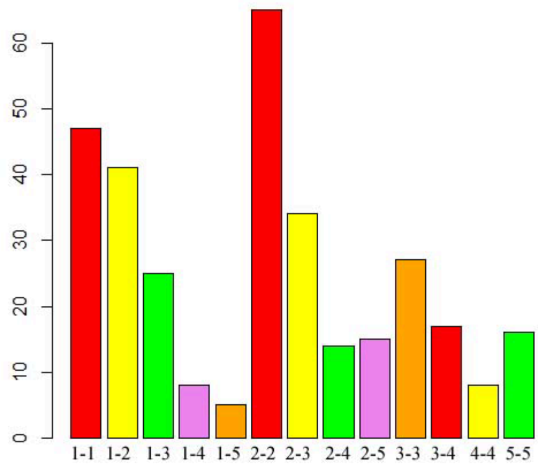


Figure 7.8: Community type for users 'future links.

To evaluate the proposed approach we use precision, recall, and F-measure evaluation

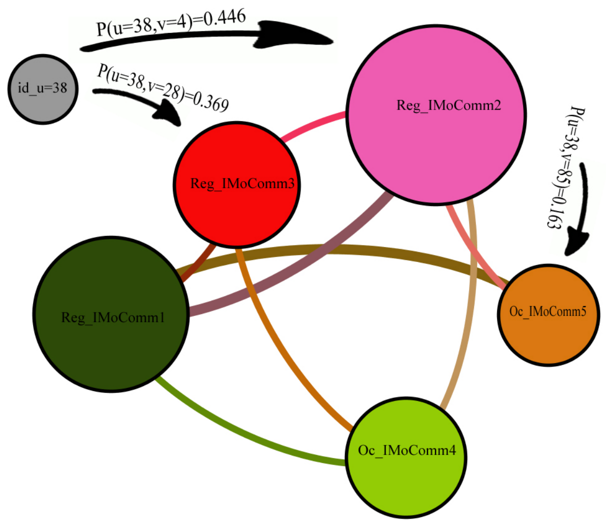


Figure 7.9: Illustartion of next IMoComms for user 38.

metric as performance measure for link prediction which is defined as follows:

$$Precision = \frac{E_r}{E_{predict}} \tag{7.4}$$

$$Recall = \frac{E_r}{E_{predict-positive}} \tag{7.5}$$

$$F - measure = \frac{2 * Precision * Recall}{Precision + Recall} \tag{7.6}$$

Where E_r, $E_{predict}$, and $E_{predict-positive}$ represent the corrected predicted links, the total predicted links, and the positive predicted links, respectively.
Our evaluation, for the three datasets shows that the average recall is 0.0.66, the average precision is 0.53, and the average F-measure is 0.59 (see figure 7.10). In *dataset1* we have an accurate prediction expressed by the recall equal to 0.66; the high precision (0.53) indicates significant prediction, thus we see that individuals exhibit regularity of belonging to their regular communities, and this community type is qualified as stable groups which appear and reappear in precise timing. The experiment with *datset2* reveals still notable prediction performance with a small decrease compared to dataset1. This is due to the formation of irregular communities during the weekend days, which generates improbable links in our model for the next users movement. Due to the presence of large number of weekends, in *dataset3* the movements of each single user do not appear as continuous. Therefore, in this case, it is more

efficient to use other users attributes extracted from their complex networks (social media, transportation networks, etc) in order to improve the prediction of the next users movement and his/her IMoComm either regular or occasional one.

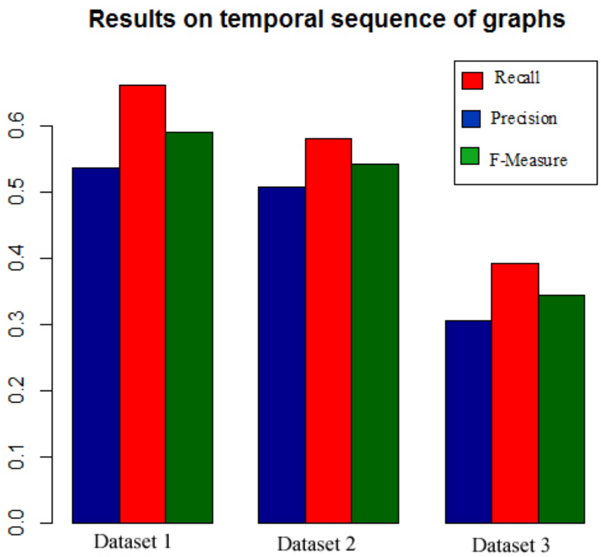


Figure 7.10: Performance of the prediction method for three social graphs during one month

The key challenge in evaluating the search space of prediction is how to train and make prediction of all possible locations of a mobile user. Existing link prediction in dynamic social networks studies focus on predicting edges between two static snapshots, this requires computing all the features for $O(|n^2| - |E|)$ node pairs. Moreover, many location recommendation algorithm for users have been proposed to minimize the computational complexity by reducing the set of candidate locations such as Location-Friendship Bookmark-Coloring Algorithm (LFBCA)[WTM13] and REGULA algorithm [MG15]. In our work, we rely on the abstract level of mobile users, due to this fact, we have made prediction in relation to dynamic mobile communities (IMoComm prediction). Therefore, the search space of prediction indicates that for each user, prediction is made by conditioning on its past history and all its daily communities. The historical IMoComm can be absorbed by a fixed times of probability matrix, while the analyzing dynamically of changes over time of this user because his/her communities members are changing. The computational complexity in this case $O(n \times l\prime \times c)$, where $l\prime$ is equal to $|G_q|$ and c is the number of user's communities, since c in practical is a small and limited number and irrelevant to the number of nodes, the running time is $O(n \times l\prime)$. It is clear that the set of candidate locations L selected from visited locations is greater than a small number of friends $l\prime$ within one temporal community. Our new approach dramatically reduces the search space for location candidates (see Table7.2).

Table 7.2: Prediction search space

Prediction Method	Computational complexity
Dynamic algorithm	$O(\|n^2\| - \|E\|)$
LFBCA algorithm	$(L^2 \times n + n^3 \times E)$
REGULA algorithm	$O(n \times L)$
IMoComm algorithm	$O(n \times l\prime)$

7.6 Conclusion

In this chapter, we have consider the problem of designing a link prediction model for location-based services. We have analyzed the dynamic of individuals at community level over different timing and thus have defined communities prediction features. We take advantage of these user's patterns and we therefore have investigated user's Interest Based Mobile Communities to reduce prediction space and then predict user mobility.

Conclusion

Recalling the thesis proposal; it focus on studying the individuals' mobility by considering the three dimensional space of: time, location and interest. Therefore, we aim to extract conclusions that can be applied to improve mobile networking applications. To confront this challenge, our contributions deal with three principle parts. Each of these works leads to several conclusions, that are summarized next.

The first part considered is the so-called topology optimization problem: nodes can self-organized cooperatively, the goal is to choose links of communication in such a way that the network is connected, and the energy-cost of the topology is minimized. In the literature, several strategies have been proposed in order to use virtual dynamic topology to get a better network organization and ensure an efficient quality of services. However, none of the existing mechanism takes into account the structure of the real-world social network that the moving users are embedded in. Thus, our ultimate goal is to construct meaningful topology based on the social aspect and the common interest of the users for the ad hoc networks; we have proposed a clustering tree topology based on structural equivalence. First, the algorithm builds a set of privileged nodes and their clusters, and thereafter , we have defined a social aware algorithm that connects only nodes in the clustering tree to nodes that are either in the same community or they are covering at least one node of this community. This process allows creating virtual backbone of in-demand community which can support an efficient broadcast process and a searching space reduction. Moreover, constructing topology is efficient to adapt quickly the routing information in ad hoc wireless networks. In an additional work, we have illustrated the interaction between topology layer and routing layer, in fact, we have deployed a routing mechanism based on topology organization, which results in fast convergence and low communication overhead.

In the second part, we seek to introduce a new mobility model that consider the social dimension and the daily individuals behaviour in order to improve mobile networking applications. However, we were struck at the beginning of the thesis by the fact that despite the presence of several studies that had found community structure in many real technological, biological, physical, and social networks, none existed has deal with discovering communities in human mobility context. For this reason, we tried to look at some available mobile data set, and develop a community detection algorithm to quantify users similarities, but after a careful analysis we realized that these data set don't provide the interesting information about the daily habits and the individuals movements in life routines such as go to work or go home. Thankfully, things are changing with the arrival of some network traces from different recent projects, some of them are not publicly like Nokia dataset, and others have been public available like Geolife dataset. Thus, we have determined which mobility-related information is Geolife dataset able to provide and their limitations [PZG+13] [PJZ+16]. Human mobility analysis requires the design of efficient algorithms and specific models, so we have proposed a mobility data mining methodology to derive more accurate conclusions about mobility. Our methodology is divided in three steps: preprocessing, discovery and analysis patterns, indeed, we have done empirical tests and retry our algorithms several times to find the suitable parameters for surrounding the social dimension. First, we have extracted the stay points, after that, we have applied density-based clustering algorithm, and we have added semantics of location to identify point of interest (PoIs). This suggests that people share common PoIs (points of interest where a user has been at), but even more they share common interests. Thus, we have mined the frequent behaviour of users with similar interests and computed the similarity of two users in terms of similar activities at similar places using dissimilarity index. In the last step, we have applied a community detection methods to find community structure. Therefore, we were able to derive a meaningful description of existing mobile communities so-called Interest based Mobile Community (IMoComm), and to represent different semantic relationships associated with users in the network.

The last part is focused on extracting community related features to predict the likelihood that the user will behave in a particular way and to define the probability of choosing a location close to his/her group of interest. To the best of our knowledge, this is the first work that analyses users mobile communities and their dynamics in human mobility research field. To achieve this goal, we have made several empirical tests on different timing: year, months, weeks, day, hours, and , indeed, theses empirical tests took more time. Thereby, we conclude that we have to model our problem with respect to different timing which are closely related to the nature of the individuals interactions and their daily activities (according to the time-line of works, food, meeting friends, ect). Thus, we have distinguished the dynamic features and the social aspects related to the evolution of IMoComm and we have proposed a novel prediction method. Our novel method incorporates useful information conveyed by users communities ties while tracking his/her mobility history which help us to increase the fraction of right predictions and reduce dramatically the space of prediction. This finding will increase the potential of wireless and pervasive applications.

Perspectives

Despite having addressed all the objectives set at the beginning of this thesis, the work carried out during the process has also opened many interesting paths to explore that can be considered as new objectives to focus on.

- Modeling the extracted IMoComms and their evolution using a suitable mathematical model which can express the dynamic of mobile communities over time and explore the snapshot timing, thus, we will be able to provide a new social model that reflects faithfully the real world people mobility.
- Improving the proposed topology with respect to the mobility features and our new social model.
- Investigating the features of the proposed topology in the protocol stack using its potential advantages to develop socially-aware wireless applications.
- Designing and deploying a software architecture based on our proposed models in order to enable available resources to take decision at real-time. This highly social context-aware architecture will provide a distinguishing features for mobile networking applications.

Bibliography

[AD] Abdellah. Boukerram Ahlem. Drif. Optimisation de la consommation d'énergie dans les réseaux mobiles ad hoc,. In *Doctoriales de Sciences et technologies de l'information et de la communication STIC'11*.

[ADGPV06] Alex Arenas, Albert Diaz-Guilera, and Conrad J. Pérez-Vicente. Synchronization reveals topological scales in complex networks. *Physical Review Letters*, 96(11):114102, March 2006.

[AFFG08] A Arenas, A Fernandez, S Fortunato, and S Gomez. Motif-based communities in complex networks. *Journal of Physics A: Mathematical and Theoretical*, 41(22):224001, June 2008.

[AG10] Thomas Aynaud and Jean-Loup Guillaume. Long range community detection. In *LAWDN-Latin-American Workshop on Dynamic Networks*, pages 4–p, 2010.

[AIS] Rakesh Agrawal, Tomasz Imieli?ski, and Arun Swami. Mining association rules between sets of items in large databases. In *Acm sigmod record*, volume 22, pages 207–216. ACM.

[AMO93] Ravindra K. Ahuja, Thomas L. Magnanti, and James B. Orlin. *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall, 1 edition, February 1993. Published: Hardcover.

[APU09] Sitaram Asur, Srinivasan Parthasarathy, and Duygu Ucar. An event-based framework for characterizing the evolutionary behavior of interaction graphs. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 3(4):16, 2009.

[APY] Charu C. Aggarwal, Cecilia Procopiuc, and Philip S. Yu. Finding localized associations in market basket data. 14(1):51–62.

[Bar] A. L. Barabasi. Linked: the new science of networks. perseus, cambridge, mass.

[Bar88] Stephen R Barley. On technology, time, and social order: Technically induced change in the temporal organization of radiological work. *Making time: Ethnographies of high-technology organizations*, pages 123–169, 1988.

[BB05] J. P. Bagrow and E. M. Bollt. Local method for detecting communities. *Physical Review E*, 72(4):046108, 2005.

[BCDL+13] Michele Berlingerio, Francesco Calabrese, Giusy Di Lorenzo, Rahul Nair, Fabio Pinelli, and Marco Luca Sbodio. Allaboard: a system for exploring urban mobility and optimizing public transport using cellphone data. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 663–666. Springer, 2013.

[BCH+11] Richard A Becker, Ramon Caceres, Karrie Hanson, Ji Meng Loh, Simon Urbanek, Alexander Varshavsky, and Chris Volinsky. A tale of one city: Using cellular network data for urban planning. *IEEE Pervasive Computing*, 10(4):18–26, 2011.

[BCH+13] Richard Becker, Ramón Cáceres, Karrie Hanson, Sibren Isaacman, Ji Meng Loh, Margaret Martonosi, James Rowland, Simon Urbanek, Alexander Varshavsky, and Chris Volinsky. Human mobility characterization from cellular network data. *Communications of the ACM*, 56(1):74–82, 2013.

[BDG+06] U. Brandes, D. Delling, M. Gaertler, R. Goerke, M. Hoefer, Z. Nikoloski, and D. Wagner. Maximizing modularity is hard. *arXiv:physics/0608255*, August 2006.

[BE] Amanda N. Baraldi and Craig K. Enders. An introduction to modern missing data analyses. 48(1):5–37.

[BE10] Amanda N Baraldi and Craig K Enders. An introduction to modern missing data analyses. *Journal of School Psychology*, 48(1):5–37, 2010.

[BHM06] Mohsen Bahramgiri, Mohammadtaghi Hajiaghayi, and Vahab S Mirrokni. Fault-tolerant and 3-dimensional distributed topology control algorithms in wireless multi-hop networks. *Wireless Networks*, 12(2):179–188, 2006.

[[Bi] Pennisi. *E (2005) Sciences*, page 309:94.

[BIL+07] S. Boccaletti, M. Ivanchenko, V. Latora, A. Pluchino, and A. Rapisarda. Detecting complex network modularity by dynamical clustering. *Physical Review E*, 75(4):045102, April 2007.

[BJ+02] Steve A Borbash, Esther H Jennings, et al. Distributed topology control algorithm for multihop wireless networks. In *Proc. 2002 World Congress on Computational Intelligence (WCCI 2002)*, pages 355–360. Citeseer, 2002.

[BK73] C. Bron and J. Kerbosch. Finding all cliques of an undirected graph. *Communications of the ACM*, 16(9):575–577, 1973.

[BKL01] Prithwish Basu, Naved Khan, and Thomas DC Little. A mobility based metric for clustering in mobile ad hoc networks. In *Distributed computing systems workshop, 2001 international conference on*, pages 413–418. IEEE, 2001.

[BKN11] Brian Ball, Brian Karrer, and M. E. J. Newman. An efficient and principled method for detecting communities in networks. *arXiv:1104.3590*, April 2011. Phys. Rev. E 84, 036103 (2011).

[BLM+] Stefano Boccaletti, Vito Latora, Yamir Moreno, Martin Chavez, and D.-U. Hwang. Complex networks: Structure and dynamics. 424(4):175–308.

[BLRS03] Douglas M Blough, Mauro Leoncini, Giovanni Resta, and Paolo Santi. The k-neigh protocol for symmetric topology control in ad hoc networks. In *Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing*, pages 141–152. ACM, 2003.

[BP10] Chiara Boldrini and Andrea Passarella. Hcmm: Modelling spatial and temporal properties of human mobility driven by users? social relationships. *Computer Communications*, 33(9):1056–1074, 2010.

[BSH03] Fan Bai, Narayanan Sadagopan, and Ahmed Helmy. The important framework for analyzing the impact of mobility on performance of routing protocols for adhoc networks. *Ad Hoc Networks*, 1(4):383–403, 2003.

[CGS+09] Ionut Constandache, Shravan Gaonkar, Matt Sayler, Romit Roy Choudhury, and Landon Cox. Enloc: Energy-efficient localization for mobile phones. In *INFOCOM 2009, IEEE*, pages 2716–2720. IEEE, 2009.

[CL03] Yuanzhu Peter Chen and Arthur L Liestman. A zonal algorithm for clustering an hoc networks. *International Journal of Foundations of Computer Science*, 14(02):305–322, 2003.

[CLC10] Ling Chen, Mingqi Lv, and Gencai Chen. A system for destination and future route prediction based on trajectory mining. *Pervasive and Mobile Computing*, 6(6):657–676, 2010.

[CMC05] Huiping Cao, Nikos Mamoulis, and David W Cheung. Mining frequent spatio-temporal sequential patterns. In *Fifth IEEE International Conference on Data Mining (ICDM'05)*, pages 8–pp. IEEE, 2005.

[CNM04] Aaron Clauset, M. E. J. Newman, and Cristopher Moore. Finding community structure in very large networks. *Physical Review E*, 70(6):066111, December 2004.

[CSN09] Aaron Clauset, Cosma Rohilla Shalizi, and Mark EJ Newman. Power-law distributions in empirical data. *SIAM review*, 51(4):661–703, 2009.

[CT00] Jae-Hwan Chang and Leandros Tassiulas. Energy conserving routing in wireless ad-hoc networks. In *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 1, pages 22–31. IEEE, 2000.

[CWLG97] Ching-Chuan Chiang, Hsiao-Kuang Wu, Winston Liu, and Mario Gerla. Routing in clustered multihop, mobile wireless networks with fading channel. In *proceedings of IEEE SICON*, volume 97, pages 197–211, 1997.

[DA05] Jordi Duch and Alex Arenas. Community detection in complex networks using extremal optimization. *Physical Review E*, 72(2), August 2005.

[DB09] Ahlem Drif and Abdellah Boukerram. Gestions des topologies dynamiques: modélisation des utilisateurs mobiles. Technical report, Université Farhat Abbes Sétif, Algérie, 2009.

[DB13] Ahlem Drif and Abdellah Boukerram. Energy efficient dsr algorithm based on topology for mobile ad-hoc network,. *International Journal of Computer Applications*, 2013.

[DB14] Ahlem Drif and Abdallah Boukerram. Taxonomy and survey of community discovery methods in complex networks. *International Journal of Computer Science and Engineering Survey*, 5(4):1, 2014.

[DBS13] Ahlem Drif, Abdallah Boukerram, and Yacine Slimani. Découverte d'une structure de communauté des usagers du web. In *4ème conférence sur les modèles et l'analyse des réseaux: approches mathématiques et informatiques, MARAMI'13,Saint-Etienne, France*, 2013.

[DBS14] Ahlem Drif, Abdallah Boukerram, and Yacine Slimani. Community discovery topology construction for ad hoc networks. In *International Wireless Internet Conference*, pages 197–208. Springer, 2014.

[DBSG17] Ahlem Drif, Abdellah Boukerram, Yacine Slimani, and Silvia Giordano. Discovering interest based mobile communities. *Mobile Networks and Applications. will appear*, 2017.

[DBSS16] Ahlem Drif, Abdallah Boukerram, Yacine Slimani, and Giordano Silvia. Can we recognize the next user's mobile community? In *International Workshop on Complex Networks and their Applications. Will appeared*, pages 1–12. Springer, 2016.

[DGG09] Allison Davis, Burleigh Bradford Gardner, and Mary R Gardner. *Deep South: A social anthropological study of caste and class.* Univ of South Carolina Press, 2009.

[DM04] L. Donetti and M. A. Munoz. Detecting network communities: a new systematic and efficient algorithm. *Journal of Statistical Mechanics: Theory and Experiment*, 2004:P10012, 2004.

[DM05] L. Donetti and M. A. Munoz. Improved spectral algorithm for the detection of network communities. *arXiv:physics/0504059*, pages 104–107, April 2005.

[DPV05] Imre Derényi, Gergely Palla, and Tamás Vicsek. Clique percolation in random networks. *Physical Review Letters*, 94(16):160202, April 2005.

[DR01] Goran M Djuknic and Robert E Richton. Geolocation and assisted gps. *Computer*, 34(2):123–125, 2001.

[[dr17] *Can we recognize the next user's mobile community? (will appaer)*. 2017.

[Dri] Ahlem Drif. Mécanisme de contrôle de topologie dans les réseaux ad hoc. In *in Embedded Systems Conference*.

[ECQ09] Nathan Eagle, Aaron Clauset, and John A Quinn. Location segmentation, inference and prediction for anticipatory computing. In *AAAI Spring Symposium: Technosocial Predictive Analytics*, pages 20–25, 2009.

[EKS$^+$] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, and others. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd*, volume 96, pages 226–231.

[EKS$^+$96] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd*, volume 96, pages 226–231, 1996.

[EM02a] Jean-Pierre Eckmann and Elisha Moses. Curvature of co-links uncovers hidden thematic layers in the world wide web. *Proceedings of the national academy of sciences*, 99(9):5825–5829, 2002.

[EM02b] Jean-Pierre Eckmann and Elisha Moses. Curvature of co-links uncovers hidden thematic layers in the world wide web. *Proceedings of the National Academy of Sciences*, 99(9):5825–5829, April 2002.

[ESMS03] Kasper Astrup Eriksen, Ingve Simonsen, Sergei Maslov, and Kim Sneppen. Modularity and extreme edges of the internet. *Physical review letters*, 90(14):148701, 2003.

[FA86] Y. Fu and P. W. Anderson. Application of statistical mechanics to NP-complete problems in combinatorial optimisation. *Journal of Physics A: Mathematical and General*, 19:1605, 1986.

[FAPV07] I. Farkas, D. Abel, G. Palla, and T. Vicsek. Weighted network modules. *New Journal of Physics*, 9:180, 2007.

[FB07] Santo Fortunato and Marc Barthelemy. Resolution limit in community detection. *Proceedings of the National Academy of Sciences*, 104(1):36–41, February 2007.

[FHR09] Hui Fang, Wen-Jing Hsu, and Larry Rudolph. Mining user position log for construction of personalized activity map. In *International Conference on Advanced Data Mining and Applications*, pages 444–452. Springer, 2009.

[Fie73] M. Fiedler. Algebric connectivity of graphs. *Czechoslovak Mathematical Journal*, 23:298–305, 1973.

[FLGC] Gary William Flake, Steve Lawrence, C. Lee Giles, and Frans M. Coetzee. Self-organization and identification of web communities. 35(3):66–70.

[FLGC02] G.W. Flake, S. Lawrence, C.L. Giles, and F.M. Coetzee. Self-organization and identification of web communities. *Computer*, 35(3):66–70, 2002.

[FLM04] Santo Fortunato, Vito Latora, and Massimo Marchiori. A method to find community structures based on information centrality. *arXiv:cond-mat/0402522*, February 2004.

[FN01] Laura Marie Feeney and Martin Nilsson. Investigating the energy consumption of a wireless network interface in an ad hoc networking environment. In *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 3, pages 1548–1557. IEEE, 2001.

[For] Santo Fortunato. Community detection in graphs. 486(3):75–174.

[For10] Santo Fortunato. Community detection in graphs. *Physics reports*, 486(3):75–174, 2010.

[Fre77] Linton Freeman. A set of measures of centrality based on betweenness. *Sociometry*, 40(1):35–41, March 1977.

[[FV] *The NS manual.*

[GA] Roger Guimera and Luis A. Nunes Amaral. Functional cartography of complex metabolic networks. 433(7028):895–900.

[GA05] R. Guimera and L.A.N. Amaral. Functional cartography of complex metabolic networks. *Nature*, 433(7028):895–900, 2005.

[GDC10] Derek Greene, Donal Doyle, and Padraig Cunningham. Tracking the evolution of communities in dynamic social networks. In *Advances in social networks analysis and mining (ASONAM), 2010 international conference on*, pages 176–183. IEEE, 2010.

[GDDG+03] Roger Guimera, Leon Danon, Albert Diaz-Guilera, Francesc Giralt, and Alex Arenas. Self-similar community structure in a network of human interactions. *Physical review E*, 68(6):065103, 2003.

[GHB] Marta C. Gonzalez, Cesar A. Hidalgo, and Albert-Laszlo Barabasi. Understanding individual human mobility patterns. 453(7196):779–782.

[GHB08a] Marta C Gonzalez, Cesar A Hidalgo, and Albert-Laszlo Barabasi. Understanding individual human mobility patterns. *Nature*, 453(7196):779–782, 2008.

[GHB08b] Marta C Gonzalez, Cesar A Hidalgo, and Albert-Laszlo Barabasi. Understanding individual human mobility patterns. *Nature*, 453(7196):779–782, 2008.

[GHJS03] Wayne Goddard, Stephen T Hedetniemi, David Pokrass Jacobs, and Pradip K Srimani. Self-stabilizing protocols for maximal matching and maximal independent sets for ad hoc networks. In *Parallel and Distributed Processing Symposium, 2003. Proceedings. International*, pages 14–pp. IEEE, 2003.

[GK98] Sudipto Guha and Samir Khuller. Approximation algorithms for connected dominating sets. *Algorithmica*, 20(4):374–387, 1998.

[GN] Michelle Girvan and Mark EJ Newman. Community structure in social and biological networks. 99(12):7821–7826.

[GN02] M. Girvan and M.E.J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, 2002.

[GPQ07] Joy Ghosh, Sumesh J Philip, and Chunming Qiao. Sociological orbit aware location approximation and routing (solar) in manet. *Ad Hoc Networks*, 5(2):189–209, 2007.

[GSPA04] R. Guimera, M. Sales-Pardo, and L. A. N. Amaral. Modularity from fluctuations in random graphs and complex networks. *Physical Review E*, 70(2):025101, 2004.

[GT95] Mario Gerla and Jack Tzu-Chieh Tsai. Multicluster, mobile, multimedia radio network. *Wireless networks*, 1(3):255–265, 1995.

[HHJ03] Petter Holme, Mikael Huss, and Hawoong Jeong. Subnetwork hierarchies of biochemical pathways. *Bioinformatics*, 19(4):532–538, 2003.

[HPYM] Jiawei Han, Jian Pei, Yiwen Yin, and Runying Mao. Mining frequent patterns without canditear generation: A frequent-pattern tree approach. 8(1):53–87.

[H.R02] H.Rheingold. Smart mobs:the next social revolution. *Cambridge, MA: Perseus*, 2002.

[HSSJ02] Zhuochuan Huang, Chien-Chung Shen, Chavalit Srisathapornphat, and Chaiporn Jaikaeo. Topology control for ad hoc networks with directional antennas. In *Computer Communications and Networks, 2002. Proceedings. Eleventh International Conference on*, pages 16–21. IEEE, 2002.

[Hun06] Marcel Hunecke. Add home. mobility management and housing project. 2006. Technical report, 2006.

[IBC+10] Sibren Isaacman, Richard Becker, Ramón Cáceres, Stephen Kobourov, James Rowland, and Alexander Varshavsky. A tale of two cities. In *Proceedings of the Eleventh Workshop on Mobile Computing Systems & Applications*, pages 19–24. ACM, 2010.

[IBC+11a] Sibren Isaacman, Richard Becker, Ramón Cáceres, Stephen Kobourov, Margaret Martonosi, James Rowland, and Alexander Varshavsky. Identifying important places in people?s lives from cellular network data. In *International Conference on Pervasive Computing*, pages 133–151. Springer, 2011.

[IBC+11b] Sibren Isaacman, Richard Becker, Ramón Cáceres, Stephen Kobourov, Margaret Martonosi, James Rowland, and Alexander Varshavsky. Ranges of human mobility in los angeles and new york. In *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2011 IEEE International Conference on*, pages 88–93. IEEE, 2011.

[IS05] Kashtan N Milo R Itzkovitz M Alon U Itzkovitz S, Levitt R. Coarse-graining and self-dissimilarity of complex networks. *Phys Rev*, E 71:016127, 2005.

[Jav08] Nagesh Javali. *Topology Control for Wireless Ad-hoc Networks*. ProQuest, 2008.

[JDY09] Jeffrey Q Jiang, Andreas WM Dress, and Genke Yang. A spectral clustering-based framework for detecting community structures in complex networks. *Applied Mathematics Letters*, 22(9):1479–1482, 2009.

[JR88] A. K. Jain and C. RICHARD. *Algorithms for Clustering Data*. Prentice Hall Advance Reference Series, 1988.

[JV02] Eun-Sun Jung and Nitin H Vaidya. A power control mac protocol for ad hoc networks. In *Proceedings of the 8th annual international conference on Mobile computing and networking*, pages 36–47. ACM, 2002.

[KB05] Ronny Krashinsky and Hari Balakrishnan. Minimizing energy for wireless web access with bounded slowdown. *Wireless Networks*, 11(1-2):135–148, 2005.

[KBCP11] Dmytro Karamshuk, Chiara Boldrini, Marco Conti, and Andrea Passarella. Human mobility models for opportunistic networks. *IEEE Communications Magazine*, 49(12):157–165, 2011.

[KF72] P.W. Kasteleyn and C.M. Fortuin. *J. Phys. Soc. Jap. Suppl. 26, 11 (1969); C.M. Fortuin and P.W. Kasteleyn, Physica*, 57:536, 1972.

[KFM+] Ann E. Krause, Kenneth A. Frank, Doran M. Mason, Robert E. Ulanowicz, and William W. Taylor. Compartments revealed in food-web structure. 426(6964):282–285.

[KFM+03] AE Krause, KA Frank, DM Mason, RE Ulanowicz, and WW Taylor. Compartments exposed in food-web structure. *Nature*, 426:282–285, 2003.

[KKKP00] Lefteris M Kirousis, Evangelos Kranakis, Danny Krizanc, and Andrzej Pelc. Power consumption in packet radio networks. *Theoretical Computer Science*, 243(1):289–305, 2000.

[KL70] Brian W Kernighan and Shen Lin. An efficient heuristic procedure for partitioning graphs. *Bell system technical journal*, 49(2):291–307, 1970.

[KMG16] Garg Kamini, Papandrea Micheala, and Silvia Giordano. Users' locations visiting prediction algorithm based on community mobility and users' interest profiling. Technical report, NetLab, ISIN-DTI, SUPSI, 2016.

[KS01] Masao Kakihara and Carsten Sørensen. Expanding the'mobility'concept. *ACM SIGGroup bulletin*, 22(3):33–37, 2001.

[LGH05] Pedro G. Lind, Marta C. González, and Hans J. Herrmann. Cycles and clustering in bipartite networks. *arXiv:cond-mat/0504241*, April 2005.

[LH14] Miao Lin and Wen-Jing Hsu. Mining gps data for mobility patterns: A survey. *Pervasive and Mobile Computing*, 12:1–16, 2014.

[LHB+01] Li Li, Joseph Y Halpern, Paramvir Bahl, Yi-Min Wang, and Rogert Wattenhofer. Analysis of a cone-based distributed topology control algorithm for wireless multi-hop networks. In *Proceedings of the twentieth annual ACM symposium on Principles of distributed computing*, pages 264–273. ACM, 2001.

[LHK+09] Kyunghan Lee, Seongik Hong, Seong Joon Kim, Injong Rhee, and Song Chong. Slaw: A new mobility model for human walks. In *INFOCOM 2009, IEEE*, pages 855–863. IEEE, 2009.

[LM04] Vito Latora and Massimo Marchiori. A measure of centrality based on the network efficiency. *arXiv:cond-mat/0402050*, February 2004.

[LN] David Lusseau and Mark EJ Newman. Identifying the role that animals play in their social networks. 271:S477–S481.

[LN04] D. Lusseau and M.E.J. Newman. Identifying the role that animals play in their social networks. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 271(Suppl 6):S477–S481, 2004.

[LN08] E. A. Leicht and M. E. J. Newman. Community structure in directed networks. *Physical Review Letters*, 100(11):118703, March 2008.

[LNK07] David Liben-Nowell and Jon Kleinberg. The link-prediction problem for social networks. *Journal of the American society for information science and technology*, 58(7):1019–1031, 2007.

[LP49] R. Luce and Albert Perry. A method of matrix analysis of group structure. *Psychometrika*, 14(2):95–116, 1949.

[LS00] Fredrik Ljungberg and Carsten Sørensen. Overload: From transaction to interaction. *Planet Internet*, pages 113–136, 2000.

[LSH08] Sune Lehmann, Martin Schwartz, and Lars Kai Hansen. Biclique communities. *Physical Review E*, 78(1):016108, July 2008.

[LW71] Francois Lorrain and Harrison C White. Structural equivalence of individuals in social networks. *The Journal of mathematical sociology*, 1(1):49–80, 1971.

[LWG10] Zhuo Liu, Bingwen Wang, and Lejiang Guo. A survey on connected dominating set construction algorithm for. *Information Technology Journal*, 9(6):1081–1092, 2010.

[LWW+04] Xiang-Yang Li, Yu Weng, Peng-Jun Wan, Wen-Zhan Song, and Ophir Frieder. Localized low-weight graph and its applications in wireless ad hoc networks. In *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, volume 1, pages 431–442. IEEE, 2004.

[LZX+08] Quannan Li, Yu Zheng, Xing Xie, Yukun Chen, Wenyu Liu, and Wei-Ying Ma. Mining user similarity based on location history. In *Proceedings of the 16th ACM SIGSPATIAL international conference on Advances in geographic information systems*, page 34. ACM, 2008.

[MBF04] Nathalie Mitton, Anthony Busson, and Eric Fleury. Self-organization in large scale ad hoc networks. In *Mediterranean ad hoc Networking Workshop (MedHoc-Net'04).*, page 0000, 2004.

[MCN11] Aarti Munjal, Tracy Camp, and William C Navidi. Smooth: a simple way to model human mobility. In *Proceedings of the 14th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems*, pages 351–360. ACM, 2011.

[MDX12] Radu-Corneliu Marin, Ciprian Dobre, and Fatos Xhafa. Exploring predictability in mobile interaction. In *EIDWT*, pages 133–139, 2012.

[Meg10] Natarajan Meghanathan. Impact of the gauss-markov mobility model on network connectivity, lifetime and hop count of routes for mobile ad hoc networks. *JNW*, 5(5):509–516, 2010.

[MG15] Steven Mudda and Silvia Giordano. Regula: Utilizing the regularity of human mobility for location recommendation. In *Proceedings of the 6th ACM SIGSPATIAL International Workshop on GeoStreaming*, pages 69–77. ACM, 2015.

[MH] Hamamache Kheddouci Mohammed Haddad. Discussion on virtual dynamic topologies in ad hoc networks. In *META'10*.

[MLH02] D Meyer, F Leisch, and K Hornik. Adaptive information systems and modeling in economics and management science. *Benchmarking Support Vector Machines*, 2002.

[Moh02] Ilyas Mohammad. *The handbook of ad hoc wireless networks*. CRC press, 2002.

[MTR12] Bivas Mitra, Lionel Tabourier, and Camille Roth. Intrinsically dynamic network communities. *Computer Networks*, 56(3):1041–1053, 2012.

[NDST14] Nam P Nguyen, Thang N Dinh, Yilin Shen, and My T Thai. Dynamic social community detection and its applications. *PloS one*, 9(4):e91431, 2014.

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

[Newa] Mark Newman. *Networks: an introduction*. Oxford university press.

[Newb] Mark EJ Newman. Mixing patterns in networks. 67(2):026126.

[New01] M E Newman. Scientific collaboration networks. II. shortest paths, weighted networks, and centrality. *Physical review. E, Statistical, nonlinear, and soft matter physics*, 64(1 Pt 2):016132, July 2001. PMID: 11461356.

[New03] M.E.J. Newman. Mixing patterns in networks. *Physical Review E*, 67(2):026126, 2003.

[New04] M.E.J. Newman. Fast algorithm for detecting community structure in networks. *Physical Review E*, 69(6):066133, 2004.

[New06] M. E. J. Newman. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences*, 103(23):8577–8582, June 2006.

[New10] Mark Newman. *Networks: An Introduction*. OUP Oxford, March 2010.

[Nex] Mark Buchanan Nexus. *Small Worlds and the Groundbreaking Science of Networks*. Norton Publishing.

[NG] Mark EJ Newman and Michelle Girvan. Finding and evaluating community structure in networks. 69(2):026113.

[NG04] M.E.J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical review E*, 69(2):026113, 2004.

[NL07] M. E. J. Newman and E. A. Leicht. Mixture models and exploratory analysis in networks. *Proceedings of the National Academy of Sciences*, 104(23):9564–9569, May 2007.

[NN12] Raj Rao Nadakuditi and M. E. J. Newman. Graph spectra and the detectability of community structure in networks. *Physical Review Letters*, 108(18):188701, May 2012.

[NP06] Mirco Nanni and Dino Pedreschi. Time-focused clustering of trajectories of moving objects. *Journal of Intelligent Information Systems*, 27(3):267–289, 2006.

[NSRD11] Anh Dung Nguyen, Patrick Sénac, Victor Ramiro, and Michel Diaz. Steps-an approach for human mobility modeling. In *International Conference on Research in Networking*, pages 254–265. Springer, 2011.

[[Pa]

[PAP+90] Simon H D. Pothen A, Liou K. P., et al. Partitioning sparse matrices with eigenvectors of graphs. *SIAM J. MATRIX ANAL. APPLIC*, 11(3):430–452, 1990.

[PBDK12] Christos Papageorgiou, Konstantinos Birkos, Tasos Dagiuklas, and Stavros Kotsopoulos. Modeling human mobility in obstacle-constrained ad hoc networks. *Ad hoc networks*, 10(3):421–434, 2012.

[PBV07] Gergely Palla, Albert-László Barabási, and Tamás Vicsek. Quantifying social group evolution. *Nature*, 446(7136):664–667, 2007.

[PDFV05] G. Palla, I. Derenyi, I. Farkas, and T. Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Arxiv preprint physics/0506133*, 435:814–818, 2005.

[PFP+07] G. Palla, I. J. Farkas, P. Pollner, I. Derenyi, and T. Vicsek. Directed network modules. *New Journal of Physics*, 9:186, 2007.

[Pim] Stuart L. Pimm. The structure of food webs. 16(2):144–158.

[Pim79] S.L. Pimm. The structure of food webs. *Theoretical Population Biology*, 16(2):144–158, 1979.

[PJZ+16] Michela Papandrea, Karim Keramat Jahromi, Matteo Zignani, Sabrina Gaito, Silvia Giordano, and Gian Paolo Rossi. On the properties of human mobility. *Computer Communications*, 87:19–36, 2016.

[PP06] Latapy M. Pons P. Computing communities in large networks using random walks. *Journal of Graph Algorithms and Applications*, 10(2):191–218, 2006.

[PSR+15] Luca Pappalardo, Filippo Simini, Salvatore Rinzivillo, Dino Pedreschi, Fosca Giannotti, and Albert-László Barabási. Returners and explorers dichotomy in human mobility. *Nature communications*, 6, 2015.

[PWJX14] Poria Pirozmand, Guowei Wu, Behrouz Jedari, and Feng Xia. Human mobility in opportunistic networks: characteristics, models and prediction methods. *Journal of Network and Computer Applications*, 42:45–58, 2014.

[PZ15] Jun Pang and Yang Zhang. Location prediction: communities speak louder than friends. In *Proceedings of the 2015 ACM on Conference on Online Social Networks*, pages 161–171. ACM, 2015.

[PZG+13] Michela Papandrea, Matteo Zignani, Sabrina Gaito, Silvia Giordano, and Gian Paolo Rossi. How many places do you visit a day? In *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2013 IEEE International Conference on*, pages 218–223. IEEE, 2013.

[R+96] Theodore S Rappaport et al. *Wireless communications: principles and practice*, volume 2. Prentice Hall PTR New Jersey, 1996.

[RAK07] Usha Nandini Raghavan, Réka Albert, and Soundar Kumara. Near linear time algorithm to detect community structures in large-scale networks. *Physical review E*, 76(3):036106, 2007.

[RB04] J. Reichardt and S. Bornholdt. Detecting fuzzy community structures in complex networks with a potts model. *Physical Review Letters*, 93(21):218701, 2004.

[RCC+] Filippo Radicchi, Claudio Castellano, Federico Cecconi, Vittorio Loreto, and Domenico Parisi. Defining and identifying communities in networks. 101(9):2658–2663.

[RCC+04] Filippo Radicchi, Claudio Castellano, Federico Cecconi, Vittorio Loreto, and Domenico Parisi. Defining and identifying communities in networks. *Proceedings of the National Academy of Sciences of the United States of America*, 101(9):2658–2663, March 2004. PMID: 14981240.

[RCSR07] Jonathan Reades, Francesco Calabrese, Andres Sevtsuk, and Carlo Ratti. Cellular census: Explorations in urban data collection. *IEEE Pervasive Computing*, 6(3):30–38, 2007.

[RLH+08] I Rhee, K Lee, S Hong, SJ Kim, and S Chong. Demystifying the levy-walk nature of human walks. *Techical Report, NCSU, http://netsrv. csc. ncsu. edu/export/Demystifying Levy Walk Patterns. pdf*, 2008.

[RM99a] Volkan Rodoplu and Teresa H Meng. Minimum energy mobile wireless networks. *IEEE Journal on selected areas in communications*, 17(8):1333–1344, 1999.

[RM99b] Volkan Rodoplu and Teresa H Meng. Minimum energy mobile wireless networks. *IEEE Journal on selected areas in communications*, 17(8):1333–1344, 1999.

[RSH+09] I Rhee, M Shin, S Hong, K Lee, S Kim, and S Chong. Crawdad data set ncsu/mobility models, 2009.

[RSH+11a] Injong Rhee, Minsu Shin, Seongik Hong, Kyunghan Lee, Seong Joon Kim, and Song Chong. On the levy-walk nature of human mobility. *IEEE/ACM transactions on networking (TON)*, 19(3):630–643, 2011.

[RSH+11b] Injong Rhee, Minsu Shin, Seongik Hong, Kyunghan Lee, Seong Joon Kim, and Song Chong. On the levy-walk nature of human mobility. *IEEE/ACM transactions on networking (TON)*, 19(3):630–643, 2011.

[S+06] J Scott et al. Impact of human mobility on the design of opportunistic forwarding algorithms. In *Proc. IEEE INFOCOM*, 2006.

[San05] Paolo Santi. Topology control in wireless ad hoc and sensor networks. *ACM computing surveys (CSUR)*, 37(2):164–194, 2005.

[SB02] Paolo Santi and Douglas M Blough. An evaluation of connectivity in mobile wireless ad hoc networks. In *Dependable Systems and Networks, 2002. DSN 2002. Proceedings. International Conference on*, pages 89–98. IEEE, 2002.

[SCCH09] Huawei Shen, Xueqi Cheng, Kai Cai, and Mao-Bin Hu. Detect overlapping and hierarchical community structure in networks. *Physica A: Statistical Mechanics and its Applications*, 388(8):1706–1712, April 2009.

[SCM+06] Jing Su, Kelvin KW Chan, Andrew G Miklas, Kenneth Po, Ali Akhavan, Stefan Saroiu, Eyal de Lara, and Ashvin Goel. A preliminary investigation of worm infections in a bluetooth environment. In *Proceedings of the 4th ACM workshop on Recurring malcode*, pages 9–16. ACM, 2006.

[SF82] Weber TA Stillinger FH. *Phys Rev A*, 25:978–989, 1982.

[SKP83] Jr. S. Kirkpatrick, C. D. Gillat and M. P.Vecchi. *Science 220, 671*, 1983.

[SKWB10] Chaoming Song, Tal Koren, Pu Wang, and Albert-László Barabási. Modelling the scaling properties of human mobility. *Nature Physics*, 6(10):818–823, 2010.

[SMD15] Yacine SLIMANI, Abdelouahab MOUSSAOUI, and Ahlem DRIF. Discovery and analysis of usage patterns for web personalization. *International Journal on Recent and Innovation Trends in Computing and Communication (IJRITCC)*, 2015.

[SMK] Carsten Sorensen, Lars Mathiassen, and Masao Kakihara. Mobile services: Functional diversity and overload.

[SMLD11] Yacine Slimani, Abdelouahab Moussaoui, Yves Lechevallier, and Ahlem Drif. A community detection algorithm for web usage mining systems. In *Innovation in Information & Communication Technology (ISIICT), 2011 Fourth International Symposium on*, pages 112–117. IEEE, 2011.

[SMLD12a] Yacine Slimani, Abdelouahab Moussaoui, Yves Lechevallier, and Ahlem Drif. Identification de communautés d'usage du web depuis un graphe issu des fichiers d'accès. In *EGC*, pages 525–530, 2012.

[SMLD12b] Yacine Slimani, Abdelouahab Moussaoui, Yves Lechevallier, and Ahlem Drif. Identification de communautés d'usage du web depuis un graphe issu des fichiers d'accès. In *EGC*, pages 525–530, 2012.

[SPGMA07] Marta Sales-Pardo, Roger Guimera, Andre A. Moreira, and Luis A. Nunes Amaral. Extracting the hierarchical organization of complex systems. *Proceedings of the National Academy of Sciences*, 104(39):15224–15229, September 2007.

[SQBB] Chaoming Song, Zehui Qu, Nicholas Blumm, and Albert-Laszlo Barabasi. Limits of predictability in human mobility. 327(5968):1018–1021.

[SQBB10] Chaoming Song, Zehui Qu, Nicholas Blumm, and Albert-László Barabási. Limits of predictability in human mobility. *Science*, 327(5968):1018–1021, 2010.

[SSZ02] Ivan Stojmenovic, Mahtab Seddigh, and Jovisa Zunic. Dominating sets and neighbor elimination-based broadcasting algorithms in wireless networks. *IEEE Transactions on parallel and distributed systems*, 13(1):14–25, 2002.

[STN15] John Steenbruggen, Emmanouil Tranos, and Peter Nijkamp. Data from mobile phone operators: A tool for smarter cities? *Telecommunications Policy*, 39(3):335–346, 2015.

[SYW+11] JB Sun, Jian Yuan, Yue Wang, HB Si, and XM Shan. Exploring space–time structure of human mobility in urban space. *Physica A: Statistical Mechanics and its Applications*, 390(5):929–942, 2011.

[TdMGP15] Jameson L Toole, Yves-Alexandre de Montjoye, Marta C González, and Alex Sandy Pentland. Modeling and understanding intrinsic characteristics of human mobility. In *Social Phenomena*, pages 15–35. Springer, 2015.

[THCS01] R. L. Rivest T. H. Cormen, C. E. Leiserson and C. Stein. *Introduction to Algorithms*. MIT Press, Cambridge, 2nd ed. edition, 2001.

[Vin] Thaddeus Vincenty. Direct and inverse solutions of geodesics on the ellipsoid with application of nested equations. 23(176):88–93.

[WF94] Stanley Wasserman and Katherine Faust. *Social Network Analysis: Methods and Applications*. Cambridge University Press, November 1994.

[WHB+12] Pu Wang, Timothy Hunter, Alexandre M Bayen, Katja Schechtner, and Marta C González. Understanding road usage patterns in urban areas. *Scientific reports*, 2, 2012.

[WLBW01] Roger Wattenhofer, Li Li, Paramvir Bahl, and Y-M Wang. Distributed topology control for power efficient operation in multihop wireless ad hoc networks. In *INFOCOM 2001. Twentieth annual joint conference of the IEEE computer and communications societies. Proceedings. IEEE*, volume 3, pages 1388–1397. IEEE, 2001.

[WPS+11] Dashun Wang, Dino Pedreschi, Chaoming Song, Fosca Giannotti, and Albert-Laszlo Barabasi. Human mobility, social ties, and link prediction. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1100–1108. ACM, 2011.

[WS98] Duncan Watts and Steven Strogatz. Collective dynamics of small world networks. *Nature*, 393(6684):440–442, June 1998.

[WTM13] Hao Wang, Manolis Terrovitis, and Nikos Mamoulis. Location recommendation in location-based social networks using user check-in data. In *Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 374–383. ACM, 2013.

[WXWZ15] Peng Wang, BaoWen Xu, YuRong Wu, and Xiaoyu Zhou. Link prediction in social networks: the state-of-the-art. *Science China Information Sciences*, 58(1):1–38, 2015.

[WZ04] Roger Wattenhofer and Aaron Zollinger. Xtc: A practical topology control algorithm for ad-hoc networks. In *Parallel and Distributed Processing Symposium, 2004. Proceedings. 18th International*, page 216. IEEE, 2004.

[YCZ+11] Tianbao Yang, Yun Chi, Shenghuo Zhu, Yihong Gong, and Rong Jin. Detecting communities and their evolutions in dynamic social networks?a bayesian approach. *Machine learning*, 82(2):157–189, 2011.

[YJY06] Zeng Yuanyuan, Xiaohua Jia, and He Yanxiang. Energy efficient distributed connected dominating sets construction in wireless sensor networks. In *Proceedings of the 2006 international conference on Wireless communications and mobile computing*, pages 797–802. ACM, 2006.

[YS] Wenjiang Pei Yi Shen. Recursive filtration method for detecting community structure in networks. *Physica A: Statistical Mechanics and its Applications*, (26):6663–6670.

[YZX12] Jing Yuan, Yu Zheng, and Xing Xie. Discovering regions of different functions in a city using human mobility and pois. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 186–194. ACM, 2012.

[Zak] Mohammed Javeed Zaki. Scalable algorithms for association mining. 12(3):372–390.

[ZH04] Lipowsky R. Zhou H. Network brownian motion: A new method to measure vertex-vertex proximity and to identify communities and subcommunities. *International Conference on Computational Science*, pages 1062–1069, 2004.

[Zha02] Yilin Zhao. Standardization of mobile phone positioning for 3g systems. *IEEE Communications magazine*, 40(7):108–116, 2002.

[Zhe] Yu Zheng. GeoLife GPS trajectories.

[Zho] Haijun Zhou. Distance, dissimilarity index, and network community structure. 67(6):061901.

[Zho03a] H. Zhou. Distance, dissimilarity index, and network community structure. *Physical review e*, 67(6):061901, 2003.

[Zho03b] Haijun Zhou. Network landscape from a brownian particle?s perspective. *Physical Review E*, 67(4):041908, 2003.

[ZKS] Max Zimmermann, Thomas Kirste, and Myra Spiliopoulou. Finding stops in error-prone trajectories of moving objects with time-based clustering. In *Intelligent interactive assistance and mobile multimedia computing*, pages 275–286. Springer.

[ZLC+] Yu Zheng, Quannan Li, Yukun Chen, Xing Xie, and Wei-Ying Ma. Understanding mobility based on GPS data. In *Proceedings of the 10th international conference on Ubiquitous computing*, pages 312–321. ACM.

[ZWL+08a] Peng Zhang, Jinliang Wang, Xiaojia Li, Menghui Li, Zengru Di, and Ying Fan. Clustering coefficient and community structure of bipartite networks. *Physica A: Statistical Mechanics and its Applications*, 387(27):6869–6875, 2008.

[ZWL+08b] Peng Zhang, Jinliang Wang, Xiaojia Li, Menghui Li, Zengru Di, and Ying Fan. Clustering coefficient and community structure of bipartite networks. *Physica A: Statistical Mechanics and its Applications*, 387(27):6869–6875, December 2008.

[ZX] Yu Zheng and Xing Xie. Learning travel recommendations from user-generated GPS traces. 2(1):2.

[ZXM] Yu Zheng, Xing Xie, and Wei-Ying Ma. GeoLife: A collaborative social networking service among user, location and trajectory. 33(2):32–39.

[ZZXM] Yu Zheng, Lizhu Zhang, Xing Xie, and Wei-Ying Ma. Mining interesting locations and travel sequences from GPS trajectories. In *Proceedings of the 18th international conference on World wide web*, pages 791–800. ACM.

[ZZZ08] Junhua Zhang, Shihua Zhang, and Xiang-Sun Zhang. Detecting community structure in complex networks based on a measure of information discrepancy. *Physica A: Statistical Mechanics and its Applications*, 387(7):1675–1682, March 2008.