Fakultät für Informatik,
Lehrstuhl 11 (Prof. Dr. Schlichter):
Angewandte Informatik / Kooperative Systeme

Ad-Hoc-Groups in Mobile Communities - Detection, Modeling and Applications

Dipl.-Phys., Dipl.-Inform. Georg Groh

Vollständiger Abdruck der von der Fakultät für Informatik der Technischen Universität München zur Erlangung des akademischen Grades eines Doktors der Naturwissenschaften (Dr. rer. nat) genehmigten Dissertation.

Vorsitzende:	Univ.-Prof. Gudrun Klinker, PhD
Prüfer der Dissertation:	1. Univ.-Prof. Dr. Johann Schlichter
	2. Univ.-Prof. Dr. Uwe Baumgarten

Die Dissertation wurde am 01.02.2005 bei der Technischen Universität München eingereicht und durch die Fakultät für Informatik am 22.04.2005 angenommen.

Kurz-Zusammenfassung

Die Arbeit beschäftigt sich mit Gruppen von Menschen, die im Rahmen einer mobilen Community kontextsensitiv mobil kommunizieren und realweltlich interagieren. Dabei ist der Begriff der Ad-Hoc-Gruppe als einer kontextuellen Manifestation oder Instantiierung existierender sozialer Gruppen von Bedeutung. Es werden im Hinblick auf die Erkennung und Modellierung von (Ad-Hoc-)Gruppen ausführlich verschiedene Beispiele (Orts- und Geschwindigkeitsdaten, natürlichsprachliche Interessen-Phrasen und hierarchische textuelle Kommunikationsinhalte) für Klassen von Daten in mobilen Communities untersucht, Ähnlichkeitsmaße konstruiert, empirisch oder über stochastische Modelle verifiziert und darauf aufbauend spezielle Clusterverfahren zur Erkennung und Modellierung von (Ad-Hoc-)Gruppen entwickelt und getestet. Eine Diskussion möglicher konkreter Anwendungen der entwickelten Verfahren schliesst die Arbeit ab.

Short Abstract

The thesis investigates groups of people in a mobile community practicing mobile communication in a context-sensitive way and also interacting in the real world. In this scenario, the concept of an Ad-Hoc-Group as a contextual manifestation or instantiation of an existing social group is of special significance. In view of detecting and modeling of (Ad-Hoc-)groups, several examples for classes of data (location- and velocity-data, natural language interest phrases and hierarchical, text-based communication-data) in mobile communities are investigated. Similarity measures are constructed and verified by empiric means or through stochastic simulation. On this basis, special clustering approaches for the detection and modeling of (Ad-Hoc-)groups are developed and tested. The thesis concludes with a discussion on possible applications for the methods which have been developed.

Contents

0 Introduction **1**

0.1 Basic Motivation 1

0.2 Thesis-Structure and Course of Argumentation 2

1 Communities and Mobile Communities **7**

1.1 General Characterizations 7

1.2 Virtual Communities 8

1.2.1 Computer Mediated Communication 9

1.2.2 Computer Mediated Communication in Virtual Communities 10

1.2.3 What Virtual Communities are not 11

1.2.4 Summarizing Characterization 12

1.3 Community Support 13

1.3.1 Requirements for Community Support 14

1.3.2 Elements of a Collaborative Information- or Knowledge-Space 15

1.3.3 Collaborative Information- and Knowledge Spaces and the Web 17

1.4 Mobility and Context 18

1.4.1 Context 18

1.4.2 Sensing Context 19

1.4.3 Using Context: Context-Aware Computing 21

1.4.4 Mobile Communities 22

1.5 An Example: The COSMOS Project 23

1.5.1 Studiosity: Analysis and Architecture 24

1.5.2 Studiosity: Web Platform and Mobile Platform 27

2 Groups and Ad-Hoc-Groups **35**

2.1 The Notion of a Group in Sciences 35

2.1.1 The Notion of a Group in Social Psychology 36

2.1.1.1 Propinquity Effects: Social Relevance of Space 38

2.1.1.2 Social Identity Theory: Self-Perception and Groups 40
2.1.2 Sociometric Descriptions for Groups 41
2.1.2.1 Undirected Graphs 42
2.1.2.2 Directed Graphs 43
2.1.2.3 Graphs with Valued Relations 44
2.1.3 Conclusions 47
2.2 Groups in Communication- and Information-Management 49
2.3 (Ad-Hoc-)Groups in Mixed-Real-World-Virtual Mobile Communities 50
2.3.1 Characterization of Ad-Hoc-Groups 50
2.3.2 Ad-Hoc-Groups in MMCs 51
2.4 Detecting and Modeling Ad-Hoc-Groups 52

3 Contextual Data: Locations and Velocities 55
3.1 Data for User-Modeling 55
3.1.1 Data for Individual User-Modeling 56
3.1.2 Data for User-Modeling in Communities 57
3.1.3 Data for Modeling Ad-Hoc-Groups in Mobile Communities 58
3.2 Accessing Localization Information 59
3.3 Privacy and Pragmatics of Location Data 60
3.4 Existing Continuous Mobility Models 61
3.4.1 Individual Mobility Models 61
3.4.2 Group Mobility Models 63
3.5 Mobility Models Used for the SUMI Simulator 64
3.6 Gauss Markov Model 65
3.7 Resting Times 72
3.8 Group Mobility Model 75
3.9 SUMI 81
3.10 Similarity Measure 82

4 Non-Contextual Data: Explicit Self Information and Communication Data with Tree-like Structure 87
4.1 Interest Phrases 87
4.2 Accessing Interest Phrases 88
4.3 Lists Of Choices 88
4.4 Lists of Free Text Elements 89
4.4.1 MTV Collection 89

4.4.2 Party Community Collection 91
4.4.3 Survey Collection 91
4.5 Similarity Measures 92
4.6 Similarity Measures for Lists of Choices 93
4.6.1 Ontologies for Lists of Choices 93
4.6.2 List-of-Choice Similarity Measure in Topic Taxonomies with Generalized Abstraction Relations 96
4.7 Similarity Measures for Free Text Interest Phrases 98
4.7.1 Spelling 100
4.7.2 Comparing Unordered Sets of Interest Phrases 102
4.7.3 Conceptual Semantic Relatedness 104
4.7.3.1 General Tools and Techniques from NLP 105
4.7.3.2 WordNet as an Example for a Semantic Net 106
4.7.3.3 Techniques for Measuring Semantic Relatedness with WordNet . 108
4.7.4 Disambiguating Word Senses 109
4.7.5 Similarity Measure for Pairs of Free Text Interest Phrases 110
4.7.5.1 The Tennis Problem Modification 114
4.7.5.2 A Remark on Inflections 115
4.7.6 The Algorithm 115
4.7.7 A Survey 115
4.7.7.1 Survey Evaluation 119
4.7.7.2 Results from Survey 1 121
4.7.7.3 Results from Survey 2 121
4.7.7.4 Results from Survey 3 122
4.7.8 Other Explict User Data 123
4.8 Communication Data with Tree-like Structure 123
4.8.1 Previous and Related Work 124
4.8.2 Test Collections 125
4.8.3 Similarity Measure 125
4.8.3.1 Vector Model 126
4.8.3.2 Comparing the Postings Contents 127
4.8.3.3 Incorporation of Discussion Thread Structure 128
4.8.3.4 Combining Structure and Content 130

5 Group Detection and Modeling Algorithms 133

5.1 Detection and Modeling of Ad-Hoc-Groups and Abstract Groups on the Basis of Spatio-Temporal Proximity and Velocity 134

5.1.1 Clustering 134

5.1.2 Crisp Clustering 136

5.1.2.1 SAHN 136

5.1.2.2 K-Means-Clustering 137

5.1.2.3 Minimum Spanning Tree Clustering 137

5.1.2.4 Crisp Cluster Validation Strategies 138

5.1.3 Basic Experiments 139

5.1.3.1 The Role of SUMI 139

5.1.3.2 Basic Algorithms 140

5.1.3.3 Socially Motivated Cluster Validation 141

5.1.3.4 Quantitative Evaluation Measure 142

5.1.3.5 Results 145

5.1.4 Detecting and Modeling Abstract Groups 150

5.1.4.1 Step 1: Similarity with respect to members 151

5.1.4.2 Step 2 Extraction of Periodicity Information 153

5.1.4.3 Fourier Analysis 154

5.1.4.4 Statistical Approach 157

5.1.4.5 Step 3: Amalgamating Ad-Hoc-Group Models into Abstract Group Models 158

5.1.4.6 Comparing Found Abstract Groups with Actual Abstract Groups 159

5.2 Detection and Modeling of Groups on the Basis of Interests 164

5.2.1 Fuzzy Clustering 164

5.2.1.1 Fuzzy C-Means 165

5.2.1.2 Fuzzy Clustering of Locations 166

5.2.1.3 Other Fuzzy Clustering Approaches 168

5.2.1.4 RACE 169

5.2.1.5 Cluster Validation for Fuzzy Clusterings 171

5.2.2 Applying Fuzzy Clustering 172

5.2.2.1 Finetuning RACE with Simulated Annealing 175

5.2.2.2 RFAO-Relational Fuzzy Alternating Optimization 177

5.2.2.3 Cluster Validation Results 178

5.2.3 Results for Interest Data 178

5.2.3.1 Free Text Interest Phrases 179

5.2.3.2 List-Of-Choice Interest Vectors . . . 181
5.3 Detection and Modeling of Groups on the Basis of Communication Patterns . . . 183
5.3.1 Test-Data and Coarse Qualitative Performance of Relational Clustering Algorithms . . . 183
5.4 Amalgamating Aspect Group-Models . . . 184

6 Applications for Group Models **187**
6.1 Indicating Groups . . . 187
6.1.1 Indicating Groups wRt. Location and Velocity . . . 188
6.1.2 Indicating Groups wRt. Interests And Communication . . . 191
6.1.2.1 Visualizing Fuzzy Sets . . . 191
6.1.2.2 Processing the Prototypes . . . 192
6.2 Collaborative Filtering Revisited . . . 194
6.2.1 Conventional Collaborative Filtering . . . 194
6.2.2 Improving CF via Group Models . . . 195
6.3 (Proactive) Context-Sensitive Information Push . . . 198
6.3.1 Location Based Information Retrieval and Filtering . . . 198
6.3.1.1 The Problem . . . 198
6.3.1.2 Location Based Information Retrieval and Filtering in Mobile Communities . . . 200
6.3.1.3 Using Group Models for Location Based Filtering . . . 201
6.3.1.4 An Example . . . 203
6.4 Group Calendar / Reminder . . . 204

7 Conclusion **207**
7.1 What was Achieved? . . . 207
7.2 Critical Discussion and Open Questions . . . 209
7.2.1 Future Prospects . . . 210

A Ontologies **213**
A.1 Examples . . . 215
A.2 Ontologies and the Semantic Web . . . 216

B Spelling Correction **217**

C Interest Test Collections and Similarity Survey **219**

Bibliography **235**

Abbreviations **249**

List of Tables **250**

List of Figures **253**

Chapter 0

Introduction

0.1 Basic Motivation

Supporting **business processes** has a long history in computer science. Over the last 10 years companies like SAP have become multi-billion dollar global players with their integrated business software bundles. Production planning, marketing, human resources, logistics, finance, resource management: There is almost no field in business engineering whose processes and data are not supported and mapped by e.g. some SAP-R3 module or comparable software.
Looking at the IT support available for **social processes**, one has to admit that no comparable solutions exist. But what has to be understood by social processes? In the case of business processes, the definition seems more obvious, since the majority of businesses aim at making money, thus a business process can be defined as an ordered sequence of actions that support the ultimate goal of making money. In case of social processes, the goals and sub-goals are usually more complex and no socio-psychologist would attempt to give a complete list. Nevertheless, common sense suggests that the most common ultimate motivation for every human being to engage in **social relations** with other humans is the benefit that social interaction means for the satisfaction of basic biological needs. Thus a social process can be defined as a correlated sequence of actions that the people involved in the process perform to create or maintain their social relations in view of collaborative goals. A simple social process is, for example, a sequence of communication acts between two people.
Just as in the case of business processes, IT can only support those aspects of **social processes that have a data-representation** in an algorithmically accessible format. A class of IT systems supporting basic social processes are communication infrastructures like e-mail systems (servers, protocols like IMAP, Mailers like Outlook etc.) or SMS systems (mobile cell phones, wireless hardware like antennae and base stations, protocols, gateways etc.). A relatively new class of IT systems supporting aspects of social processes in certain defined groupings of people are **virtual community support systems**. A virtual community can be seen as a set of people sharing a common pursuit (e.g. a common interest) that implies communication via IT systems. If this communication implies the use of mobile devices, these groupings can be called **mobile communities** or mixed virtual mobile communities. If the mobile communication is also included in the integrated support system for these special communities, we can speak of support systems for mobile communities. **COSMOS**, a scientific project at the TU München [8], aims at investigating the technical, social and organizational aspects of such IT systems.

The social processes supported by a such a system for mobile communities can in essence be modeled as **communication processes**. Information is transferred between members of the mobile community with the general goal of maintaining their social relations. What are the main actors communicating with each other with the help of such a system and what are the interaction situations like?
Usually, members of a conventional virtual community interact with a community support system in an isolated situation in front of their desktop computer. In a mobile community, the **interaction situations** are much deeper embedded into the all day life of the community members and are much more manifold. Thus the spectrum of interaction situations is also much more diverse. Interaction situations that are characterized by an increased number of dynamic context parameters (such as situations occurring in narrow time-windows or around special locations) are the rule rather than the occasion and a mobile community support system must face the challenges associated with context-sensitivity: modeling of context, new privacy problems, special user-interface development etc.. A more diverse spectrum of interaction situations also brings about a more diverse spectrum of different **actors**. In the usual desktop interaction paradigm an interaction situation that directly involves **groups** is a rather rare case. Software supporting groups (like typical groupware or community-support systems) aims at groups that are rather permanent and do not change very dynamically like e.g. a team of software developers. The groups and services supporting them may be strictly defined and oriented on concrete tasks (teams) or may be more fuzzy and flexible (communities).
In a mobile community support system, **groups that are highly dynamic** may also become actors (e.g. as sources or destinations for mobile communication services). These highly dynamic groups will be called **Ad-Hoc-Groups** in this thesis. They may or may not be instantiations of more stable permanent social groupings (called abstract groups), may occur in special contexts only and may vary with respect to their members. The deep embedding of mobile devices and mobile community support in the lives of the users makes it possible to collect data that reflect the formation and the actions of such Ad-Hoc-Groups while conventional community support only allows for data that reflects the formation and actions of longer lasting groups.

In this thesis we will investigate and develop methods that allow for the detection and the modeling of such Ad-Hoc-Groups and their underlying abstract groups in mobile communities and we will investigate possible applications for these models.

0.2 Thesis-Structure and Course of Argumentation

Chapter 1

In chapter 1 the concept of **community and community support** will be shortly reviewed since communities act as a background framework for the group detection and modeling procedures discussed in the thesis and because they can be perceived as basic groupings themselves. We will investigate and work out basic characterizations and develop basic vocabulary for the later discussion. The main sections of chapter 1 discuss the special case of **mobile communities**. Starting from basic characterizing definitions in connection with mobile communities and from a short review of existing work on context sensitivity and its relevance for mobile communities we proceed to the discussion of a concrete case study: The **Studiosity community** of the **COSMOS project**, which we will use to exemplify the preceding notions and discussion.

Chapter 2

In order to further prepare the more formal and algorithmic topics of the following chapters, chapter 2 steps back one step and investigates groups from a more abstract point of view but also discusses aspects of interdependence of groups and context. Chapter 2 first introduces the **notion of a group** from the point of view of other scientific disciplines such as **socio-psychology** and mathematical psychology. After dealing with basic approaches for characterizing and modeling groups, the special meaning of the context parameter location for the formation and psychological perception of groups is discussed and socio-psychological theories supporting the thesis that **spatial proximity** is beneficial for the formation of groups with certain characteristics are discussed. From these theories we then derive arguments why context parameters such as location are also good indicators for groups. As a further central point of this chapter we define the **notion of an Ad-Hoc-Group**, which is basically a group which forms in a distinct spatio-temporal situation and for which e.g. co-location is a good indicator.

Chapter 3

The next two chapters are devoted to the investigation of classes of **data sources** that can be used to detect Ad-Hoc-Groups and to the development of similarity measures on these data which can be used as an input to the detection algorithms introduced in chapter 5.
Chapter 3 starts with an overview of basic classes of data that are used to infer user-characterizations. The more special case of data in collaborative information and knowledge spaces of communities in view of user-characterization is discussed after that. Even more special is the case of context-data in mobile communities which is reviewed and discussed next. The basic distinction that is made in view of chapters 3 and 4 is between data with **fast dynamics (context data)** which are investigated for the rest of chapter 3 with location and velocity as the main example and data with **slow dynamics** such as **interests and communication data** that are subject to chapter 4. From the set of highly dynamic data, **location and velocity** play a special role which is emphasized in the further course of chapter 3. Since location and velocity data are very sensitive with respect to privacy, no real experiment data is available to test the algorithms on. Thus the investigation of location and velocity as basic data for the detection and characterization of Ad-Hoc-Groups implies that a suitable **realistic stochastic simulation** has to be developed. The main section of chapter 3 is devoted to the discussion of various aspects of the development of this simulation which involves a rich spectrum of stochastic models which are necessary to produce sufficient realism. What is completely new is that in contrast to conventional mobility simulation models, the **SUMI model** developed here is capable to treat individual motion on equal footing as motion of groups. This is necessary because we now have a simulation at hand which allows for quantitative evaluation of the algorithms in chapter 5. Even highly accurate and complete realistic movement data could not provide the details about the underlying "real" group structures which is possible through SUMI. The chapter ends with a discussion on similarity measures for location and velocity data which are needed for the detection algorithms.

Chapter 4

Chapter 4 deals with data with a slower dynamics and less context-sensitive character: Interests and communication data with tree-like structure. **Interests** are in on form or the other part of

user-profile-data in many communities. What makes them so attractive for user-characterization and group characterization is that they are **explicit self descriptions** of the user while the other classes of data are implicit. We first discuss two basic types of interest representations and investigate examples for both cases. The first class are **sets of free text interest phrases** which have a special linguistic structure and have a high expressiveness and the second class are **vectors of pre-formulated interest keywords or -phrases** that are chosen from a given taxonomy (list-of-choice interest vectors). The main section of the first part of chapter 4 is devoted to the development of **similarity measures** for the two classes of interest data that allow to **compare persons** with respect to their interests. For the first class, techniques from Natural Language Processing are used and adapted to express semantically rich relations between all levels of the free text phrases, from the word-level to the phrase level which finally lead to the development of a similarity measure. In case of the list-of-choice vectors another similarity measure is constructed. Both measures are tested quantitatively by comparing the results of the similarity calculations on specially collected test-sets with the results of a survey where human participants were asked to rate the similarity of the interests and interest parts. The second part of chapter 4 is devoted to discuss another important class of community data with slow dynamics which allows for implicit user-characterizations: **Textual communication data with a tree-like structure**. This type of data occurs very frequently in community support applications as threaded discussion boards where statements and replies by the users are organized as a tree. By integrating content analysis techniques and an approach that repects the strength of social ties through analysis of reply frequencies, a similarity measure is constructed that allows to compare two users with respect to their communication behavior.

Chapter 5

Chapters 3 and 4 prepare the field for the detection and modeling of Ad-Hoc-Groups and abstract groups while chapter 5 discusses the actual **detection and modeling algorithms on the basis of the similarity measures** and simulations developed before. The first part is devoted to the detection and modeling of Ad-Hoc-Groups. As an example, **detection of Ad-Hoc-Groups with respect to locations and velocities** is discussed in great depth. The basic approach is to use adapted clustering algorithms that respect the socio-psychological subtleties of Ad-Hoc-Group formation. Furthermore, approaches for the detection and modeling of the underlying abstract groups are investigated. The Ad-Hoc-Group and abstract group models resulting from applying the developed algorithms to the SUMI simulation data are **quantitatively compared** against the simulation to allow for an estimation of the quality of the algorithms. The second part discusses the detection and modeling of **groups with respect to interests and communication data**. The nature of these classes of data makes it necessary to introduce approaches on the basis of relational **fuzzy clustering algorithms** which also have to be adapted to the purposes of group detection and modeling. The chapter is ended by a discussion on how to **combine group-models** computed with respect to different types of data.

Chapter 6

This chapter is devoted to the possible applications for the group models whose algorithmic generation was topic of the main part of the thesis. Three application fields are suggested. The

first application is indication of the group models. Arguments are discussed why indication in textual form or in other visualization forms can be beneficial as means of enhancing the communication in mobile communities. The second field of applications is collaborative filtering on the basis of group models. Traditional Collaborative Filtering is investigated and the improvements that the introduction of group models can mean for this class of services are debated. Finally, the benefits of group models for a third class of applications is discussed: Context Sensitive Information Filtering. Again we discuss the service class considering location based filtering as an example.

The thesis is concluded by a summary and a critical discussion of the results.

Chapter 1

Communities and Mobile Communities

This chapter discusses virtual communities from several points of view. The first two sections are devoted to the notion of a (virtual) community. They start by briefly summarizing the characterizations and definitions of previous work about communities. Communication acts are then identified as a basic model for actions in communities and special features of community communication are considered. A summarizing characterization for the concept of a community within the focus of this thesis is given. In the third section, a closer look will be taken at the requirements for supporting communities and collaborative information- and knowledge-spaces are identified as key components of communities. A model for such spaces is presented. The next part of this chapter is devoted to aspects of mobility and context-awareness. The notions involved are shortly reviewed and the influence of mobility and context on communities and community support are investigated. The chapter results in a brief introduction into the COSMOS project conducted at the Technische Universität München which is targeted towards applied and prototypical research on mobile communities.

1.1 General Characterizations

The word **community** can etymologically be traced back to the Latin Nomen **communitas, (-tatis)** which is essentially semantically equal to **community**, or to the adjective **communis, (-is)**, respectively, which means **shared by all or many** and which is a composition of **com-** (which means **together**) and **munia, (-orum)** (pl.) (which means **public duties**) [123, 64]. Our modern perception of the concept(s) behind the word community may differ from that of the ancient Rome, yet there are so many concepts and perceptions linked with the term community that an attempt to cover them all and from every scientific point of view does not make sense. Even within one scientific discipline (such as social science or economic science) there is no precise definition of the term. Therefore it has to be pointed out explicitly what type or class of community is the subject of discourse. The **most general class of community** dealt with in this thesis is a set of people characterized by the following parameters:

- Community members share an **awareness** of being a member in a particular community. Community awareness is a state of mind that goes beyond mere intellectual perception of

a factual state. It also includes an emotional tie to the community and is connected with the will to be a part of it [183, 76, 122, 87].

- A dense net of social relations exists among the members with a special emphasis on **communication** relations [34, 87, 88].
- Community Members have a **common pursuit** which implies a motivation to actively participate in the community [34, 87, 24, 86] and which also implies a set of rules or conventions within the community [185, 168, 87].
- Community members share one or more **similar personal parameters** where common location of living or working and common interests are prominent examples. This gave rise to characterizations or terms such as **community of practice**, **community of interest** etc. [23, 169] (see e.g. [86] for more details and references).

The spectrum of the community member's actions which are causally connected with characteristic parameters of the community (e.g. the community pursuits) is very broad. It can range from physical actions (crafting some object) to communication acts of all sorts and using all kinds of media (paper-fanzines, physical blackboards, telephones etc.).

1.2 Virtual Communities

With the advent of modern communication means (e.g. the Internet), common physical parameters for communities such as common location became less important and communities emerged, where the members did not know each other "physically". In 1993, Rheingold created the term **Virtual Community**. In his book [158] he drew a connection from e.g. early forms of newsgroups, where preliminary stages of collaborative information spaces could be observed, to the concept of a community.

The essential restriction of a pure virtual community with respect to the characterization of the general class of communities in the previous section is that all community-interactions are conducted via electronic media [23, 87] or, more strictly, conducted via networked computer-systems. Interaction is limited to the transfer of digital data. Therefore all actions in a pure virtual community will be modeled as communication acts.

Communication acts can generally be classified according to several ordinal dimensions such as

- **Synchronicity** [fully synchronous (e.g. talking face-to-face) – fully asynchronous (e.g. a letter)]
- **Direction** [direct (e.g. an explicitly addressed e-mail) – indirect (e.g. a blackboard posting)]
- **Cardinality** [1:1 (e.g. personal communication) – n:m (e.g. communication between groups)]
- **Anonymity** [anonymous (e.g. spam mail) – non-anonymous (e.g. e-mail where both parties know each other)]

and also according to several nominal dimensions such as involved **senses** (visual, acoustic, tactile, etc.), **form** (text, spoken language, gestures, etc.), and **medium** (face-to-face, paper, networked computers, etc.).

So how does communication in a virtual community differ from communication in a "real world community"? Strictly speaking, the use of electronic (digital) media or networked computers, respectively, is not a fully sufficient criterion to distinguish between the two classes. On the one hand, typical real world communication like a telephone-conversation can be conducted via electronic (digital) media or protocols e.g. ISDN or Voice-over-IP. On the other hand, a technologically advanced virtual reality-infrastructure with virtual reality displays, sensors and physical actuators can, in principle, realistically emulate a real world communication interaction. Thus, a sharp distinction surely cannot be made.
In order to characterize the differences between communication in a virtual community and communication in a "real world community" one will first have to look at the characteristics of Computer Mediated Communication.

1.2.1 Computer Mediated Communication

Computer Mediated Communication (CMC), in general, can be compared to real world communication (RWC) in terms of the aforementioned classification dimensions (see [31] for a more detailed discussion):

- In CMC, **asynchronous or semi-asynchronous** forms of communication are predominant (E-Mail, News-Groups and Discussion-Boards etc.) whereas in RWC, at least in communities, usually synchronous forms of communication are preferred (face-to-face, telephone, etc.). Semi-asynchronous communication, where communication partners tolerate reaction times from several seconds (chat-conversation) to several minutes or several hours (reply to an e-mail or a discussion board posting) exclusively exists in CMC.

- While RWC can use the complete sensual and formal spectrum, in CMC **text-based** communication is still the most wide-spread form, so that in socio-psychological literature it is consensus to use the term CMC only in relation to text-based communication [31]. While confinement to text-based communication may represent a severe limitation or constraint (Channel-Reduction-Model; see [31]) it also creates new possibilities of expression (Imagination Model and Social Information Processing Perspective [187]; see [31]). These new possibilities involve Emoticons or textual expressions for sensual impressions.

- CMC has a high percentage of **indirect communication and communication that involves indirection**: A discussion board may contain postings which have the character of a reply and are directed to one person. Nevertheless, the author and recipient are aware of the public readability of the board and other people may also access and profit from the posting.

- CMC is much more tolerant towards **anonymous communication or alternative identities** than RWC. The reduction in expressiveness in CMC leads to a reduction of socio-normative signals and thus to a decreased inhibition threshold towards socially sanctioned behavior such as anonymous utterances (Social Filter models; see [31]). This may have positive as well as negative effects [87]. Furthermore, expressiveness-reduction also offers

new degrees of freedom in self-portrayal which allows for a greater spectrum of roles and identities to be used in CMC (Simulation Model; see [31]).
In text-based CMC, personal profiles are a means to compensate for missing sensual information about a person [87].

1.2.2 Computer Mediated Communication in Virtual Communities

Compared with general CMC, CMC in a virtual community is characterized by further specializing factors. in [189], Watzlawick emphasizes the importance of social relations in communication in his axioms of communication theory. Axiom two reads:

> Jede Kommunikation hat einen Inhalts- und einen Beziehungsaspekt, derart, dass letzterer den ersteren bestimmt und daher eine Metakommunikation ist.
> (Every communication has a content-aspect and a relation aspect such that the latter determines the former and thus represents a meta-communication)

In CMC, the relation aspect has to be supported and emulated with the help of profiles, emoticons and the like. In a virtual community, communication is much more determined and augmented by the net of social relations among community members which are usually more explicitly represented than in CMC in general. Thus communities can be expected to provide a richer communication environment than other CMC environments.

In section 1.1, a common pursuit and similar personal parameters of community members have been have been mentioned as characteristic features of a community.
In virtual Communities of Interest [23], the common personal parameter is an interest in some field of information or field of knowledge, and the common pursuit is to collaboratively construct an information- or knowledge-space which thematically reflects the domain of interest.
In case of virtual Communities of Purpose [23] where the community's pursuit is quite explicit and a high degree of awareness about this pursuit or ambition exists among the community members [86], the build up of a **Collaborative Information- or Knowledge-Space** (CIKS) can be regarded as an essential means to reach the "goal". Although the ultimate "goal" is not the build up of a CIKS alone and although the "goal" cannot be reached by a CIKS alone, a CIKS is a key means to keep the members well informed and able to e.g. competently argue in public or political discussions in order to work towards the goal. An analysis of other types of (virtual) communities [23, 86, 169] also shows that a CIKS is a key part of the community's pursuit. Strictly speaking, within the limitations of a virtual community, the build up of a CIKS is all that can be achieved in view of a community's pursuit. Thus it is reasonable to identify the common pursuit of a virtual community with the pursuit to collaboratively construct an information- or knowledge-space.
In the present context, a collaborative information space is informally characterized as a set of digital data which can be interpreted as a set of information and the main purpose of which is communication between humans. The collaborative information space needs to have a structure and methods that allow for the dissemination of information (accessing, searching, browsing etc.) and the collaborative input of information. Knowledge is informally characterized as operational information. If the information in the information space can be characterized as being operational (if it can be directly used to solve some problem) the information space can be characterized knowledge space. In this model, every action (communication act) in the community changes the state of its CIKS.

As a consequence, CMC in virtual communities does have a strong bias towards having an n:m cardinality, because collaboration in input involves more than one person and, in general, more than one person will access the resulting parts in the information- or knowledge space. Furthermore indirect communication and communication that involves indirection is even more predominant than in general CMC.

1.2.3 What Virtual Communities are not

As a conclusion of the characterization of virtual communities, we will now distinguish concepts and notions associated with virtual communities from similar or adjacent concepts and notions in other fields of computer science.

Collaborative Information- or Knowledge-Space vs. Knowledge Base The concept of a collaborative information- or knowledge-space (CIKS) associated with a virtual community must be distinguished from the concept of a Knowledge Base in Artificial Intelligence. Although that concept is not precisely defined in Artificial Intelligence, it can be characterized as a declarative set of statements in a formal logic based language, representing facts, rules, and other knowledge-representation entities and which can be used for automatic problem-solving e.g. for deducing new knowledge, for autonomous agents to decide about actions etc. (see [58] for a detailed discussion). No matter what precise definition for Knowledge Base is used, a Knowledge Base always has the property of containing formal representations of explicit knowledge. Formal-explicit knowledge is knowledge that can be directly used for automatic problem-solving without expert intervention or human intervention at all. In contrast to that, human experts often organize their knowledge in an implicit way. This involves a mixture of knowledge at different levels of explicitness and formalization, emotions and other cognitive structures and processes which are not fully understood scientifically. [58] It is often even difficult for a human expert to externalize or "explicify" his knowledge in human readable (semi-formal) form as drawings, texts, etc. so that non-experts can use this semi-formal-explicit knowledge directly for problem-solving. The compilation of semi-formal-explicit knowledge into formal-explicit knowledge is subject to Expert-System research, a subfield of Artificial Intelligence.
Figure 1.1 shows the various degrees of formalization and explicitness of knowledge and information of typical entities discussed above.

A CIKS of a virtual community contains knowledge in most cases in semi-formal-implicit or in some cases in semi-formal-explicit form. The barriers to provide information or knowledge in an implicit way as opinions, thoughts, emotional statements etc. within a community are much lower than the barriers to publish information or knowledge in a more formal context such as a scientific paper where the demands in terms of explicitness and logic consistency are much higher. In a community, questions of trust and other social factors play an important role in publishing and evaluating the information or knowledge. The informal nature of a community allows for information to be published or communicated which is not well explicified or is hard to explicify at all, while at the same time a thematic focus is kept and social relations between community members allow for the evaluation of the information or knowledge. These properties of CIKSs in virtual communities make them valuable tools from an information- or knowledge management perspective because these spaces represent a "missing link" between informal implicit representations which are very intimately linked with humans and hard to access and the formal explicit representations of knowledge bases which are extremely difficult to build.

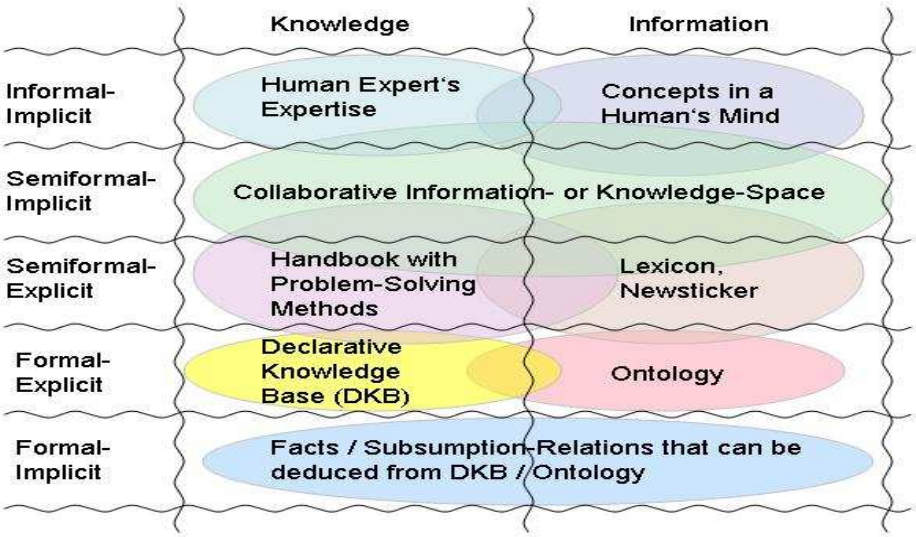


Figure 1.1: Some examples of knowledge and information in various degrees of explicitness and formalization. Regard that neither boundary is sharp.

Warm vs. Cold Since semiformal information or knowledge in a community is often implicit it is usually tightly bound to the persons associated with that information or knowledge such as the author(s), the recipient(s), etc.. In [87] this type of information or knowledge is called "warm". This is in contrast to other forms of published semiformal information or knowledge such as an online encyclopedia like Microsoft Encarta which is edited by an editorial staff and which is usually much more explicit. This type of information or knowledge can accordingly be called "cold".

Community vs. Team Another distinction has to be made between a community and a team [16, 87]. A team is usually much smaller in extension than a community, has a clearly defined structure and has a clearly defined goal. The purpose of the team is to reach the goal by collaboratively working on common items such as code-repositories, documents etc.. In order to do that, the team-members have well defined roles and duties. A typical example for a team are programmers and software architects working on a software project. In contrast to that, a community has a pursuit which is usually not that clearly defined as the goal of a team. Furthermore the structure of a community is more fuzzy: roles and duties are often not formally defined and the counterpart of the team's common working items is the community's CIKS.

1.2.4 Summarizing Characterization

A summarizing characterization for the concept of a virtual community will now be given in order to sum up what has been said about communities and virtual communities in the previous sections and in order to qualify as precisely as possible, what types of virtual community will be dealt with in this thesis. We will focus on communities that match the following characterization:

A set of people which have a high degree of community-awareness, communicate with

> other members via electronic media, and have a common pursuit which can be identified with the pursuit to collaboratively build up a thematically focused, information- or knowledge-space. This collaborative information- or knowledge-space (CIKS) predominantly contains semi-formal implicit "warm" information or knowledge with a strong emphasis on textual form.

As has been pointed out before, this characterization is still broad enough to cover a great deal of the virtual communities on the Web today. From now on, the adjective "virtual" will sometimes be dropped if it is clear from the context what type of community is meant.

1.3 Community Support

Starting from the upper characterization, supporting the common pursuit of a community is equal to supporting the management of a community's CIKS. Before discussing the basic aspects of community support, one needs to clarify some notions. We have to distinguish between

- **the community as such** (a set of people). It can be a real-world community, a virtual or a mixed real-world/virtual community. Such mixed communities will be dealt with in section 1.4.4.
- **the data representation of a virtual community**. This involves all the data created and manipulated by community members in relation to the community's pursuit and is equal to the data representation of the community's CIKS.
- **a theoretical model of a community**. We will deal with several models for certain aspects of a community in this thesis.
- **algorithms or methods for community support** that work on the data representation of the community's CIKS utilizing these theoretical models.
- **their implementation in form of software**. Often several algorithm-implementations for community support together with data-structures for elements of the community's CIKS are bundled together to form an integrated community support system. See [87] for examples and an in-depth discussion.
- **a concrete instance** of these implementations together with the data representation associated with a particular community. In [87] this is called a Community Platform. Several examples for communities and their platforms will be presented in section 1.5.

Listing these notational facets in connection with the perception of the phenomenon "community" is intended as a basis for a precise discussion. Not all of the concepts behind these notational facets will be dealt with in detail in the course of the thesis

Software for community support exists since the early days of the Internet although its use in the context of communities might not have been foreseen or planned by its creators. Basic classes of existing software are mentioned in [87].
Chat-Systems are means for text-based semi-asynchronous n:m communication which involves indirection. ICQ [31] is an example for a large chat-infrastructure that brought about a large

variety of thematically or regionally focused chat-"channels" whose members well adhere to the definition of a virtual community.
News-Groups are also thematically focused and represent another means for semi-asynchronous to asynchronous n:m communication with an even stronger involvement of indirection. People posting in a certain newsgroup also can be viewed as a virtual community.
Buddylist-Systems such as AOL Instant Messenger are a means for semi-asynchronous n:m communication where the domain of indirection is limited to a list of buddies. This is interesting because it is a means to restrict communication to a certain circle of people and is one of the most important direct representations of social structures in a virtual community.
Matchmaking and recommendation systems support the usability of a CIKS by matching people with people or information with people. Matchmaking is prominent in virtual dating communities such as friendscout24.de where personal profiles are key elements of the CIKS and are matched against each other. A well known example for a recommender system is integrated in the Amazon E-Commerce bookshop. Browsing and purchase histories of the customers together with online recensions of items can be viewed as a CIKS which is taken as a basis to recommend books or other items to users.
Many knowledge management systems integrate elements of informal communication comparable to a CIKS, because of the "missing link" nature of a CIKS that has been mentioned before. Integrated systems for virtual community support include e.g. the Cassiopeia Community Application Server which will be shortly discussed in section 1.5, the CoBricks framework [87] or the building block collection PHP-Nuke (www.phpnuke.com).

1.3.1 Requirements for Community Support

The possible detailed requirements for community support are as manifold as the spectrum of communities itself. In [87] many such requirements and solutions for these requirements are discussed.
Here, we have restricted ourselves to communities whose pursuit can be identified with the pursuit to collaboratively build up an information- or knowledge-space and whose actions are communication acts which change the state of the CIKS. Thus, on this level of abstraction, requirements for community support are requirements associated with the management of a CIKS or requirements for communication support, respectively.
We will give only a short overview of these requirements. Some aspects will be discussed in greater depth in later sections, many others have already been extensively discussed elsewhere [87, 86].

Support for Direct Communication Direct communication is an important subfield of communication. The more direct communication channels are available the richer is the expressive spectrum for the community's members. Usual channels for non-anonymous, direct, text-based (semi-formal) communication are 1:1 asynchronous (e-mail), 1:n asynchronous (group-e-mail, mailing-list), 1:n semi-asynchronous (chat-room, instant messenger) or 1:1 semi-asynchronous (private chat-room, instant messenger) channels.

Besides the mere provision of these channels, supporting direct communication needs to provide means for an integrated management. This includes e.g. Reachability Management, which regards the user's current context and chooses an appropriate channel automatically or aids the user in choosing an appropriate channel [88]. It also includes management of the exchanged

content of direct communication under the paradigm of the collaborative information- or knowledge space (CIKS). This involves searching the messages, topically classifying messages, mapping message categories [51, 97, 96] and many more. In other words, what is required is supporting the communicating parties in handling and managing the content of their communication. Furthermore, aspects of indirection can be added by allowing other community members access to these contents. These additional management requirements make direct communication in a community special because the bundle of managed communication channels needs to be fine-tuned to the needs of the specific community to provide optimal support.

Support for Indirect Communication It has been discussed before that indirect communication plays a very important role in communities. Examples for indirect, non-anonymous, text-based (semi-formal) communication are 1:n asynchronous channels (News-Groups, discussion boards, blackboards) and n:m asynchronous channels (Wikis [130], collaboratively editable web-sites). Supporting indirect communication involves means to collaboratively change, add or delete information or knowledge in form of communication content. It also includes management of content-user relations (collaborative filtering, recommendations, bookmarks, topic-maps, information retrieval). Furthermore, the management of user-user relations needs to be supported in form of matchmaking, expert-finding, buddylists, access-control etc.. The user-user relations are key components of any form of communication because they allow for the evaluation of the communicated content and provide a social context for it.

1.3.2 Elements of a Collaborative Information- or Knowledge-Space

In this section we will take a closer look at the constituting elements of a CIKS and provide a general structural model for these elements. This general structural model is a result of several years of experience in community support system design (see e.g. [88, 87, 53, 49]). The model will be restricted to textual data because an extension to graphic data, video or audio would be beyond the scope of this thesis.
In this model, the **collaborative information- or knowledge space** of a community is being constituted by user-profiles, information- or knowledge-items and relation-objects.

User Profiles Users make up a virtual community. Since in a pure virtual community there is no physical interaction, a data representation of a person is necessary to partly compensate for the complete sensual impression of a person and the personal knowledge about this person. This data representation of a user is called a user profile. Each profile corresponds to one identity of a person.

> A **user profile** is a textual data-representation of an identity of a person and the immediate environment which is in direct relation to that person. It is a subset of data representations of all available information about the person.

Available means that it needs to be representable in a computer and needs to be accessible to a computer (e.g. measurable by a sensor). The immediate environment includes the immediate physical environment (e.g. temperature, humidity, locations of nearby W-Lan access points etc.) and the immediate virtual environment (e.g. the list of applications that have been used over the last three weeks, the color of the screen background etc.).
User-Profiles contain rather static information such as name, age, address, long-term-interests,

etc. and rather dynamic information such as location, emotional state, short-term-interests, etc.. Services with a high degree of incorporation of static user profile information are often referred to as services with a high degree of **personalization**. The incorporation of dynamic user profile information is often referred to as **context-sensitiveness** (see section 1.4)
User profiles may also contain meta data (such as date of last modification). Besides textual data, user profiles will typically also contain other forms of data such as portrait-images which will not be considered here.
Usually, profiles are formally modeled as sets of attribute value pairs, where attributes are complex types, classes or concepts and values are instances, objects or elements.

Information-or Knowledge-Items The content of communication between the community members can be quantized into Information-or Knowledge-Items. For the sake of brevity, we will sometimes refer to them shortly as information items only.

> An **Information- or Knowledge-Item** is a textual data-representation of a quantum of information or knowledge that is focused to one topic or a small number of topics and to which a set of meta-data can reasonably be attached. This set of meta-data is also part of the item. Information Items are usually rather limited in extension (length and number of communicated aspects).

The main reason for quantizing the content into such items is that assigning meta data to content with a defined scope is more easy this way and that such items can be handled more conveniently by algorithms. Furthermore, the fact that the content which circulates in a community is the content of communication often introduces a natural quantization (postings in discussion boards, messages, Wikis etc.). The implicit, "warm" and fuzzy nature of the content mould the language and style of information items.
Meta data for items includes all information that is directly linked with the item e.g. elements like time of creation, time of last modification and size.
If the content is considered a special attribute, items can also be formally modeled as sets of attribute value pairs.

Relations Information-Items and User-Profiles are related to each other in various ways. A good example is the authorship relation that connects an information item with a user.

> **Relations** are data-objects that represent unary, binary and n-ary relations either between information-items, between users and information items or between users. Relations also incorporate meta data such as the algebraic property of a relation (transitivity, reflexivity etc.) or the strength of a relation.

Relations are very important structural entities because they are the glue between the other entities that decides about the usefulness of the community's CIKS.

- **User-user relations** will often be referred to as social relations. Formal models for user-user relations and their distributed management and applications are discussed in [46]. Group relations are unary social relations which will play an important role in later chapters.
- **User-item relations** include ratings used in collaborative filtering or recommender-systems, or sender-content and recipient-content relations for items which represent the content of direct communication.

- unary **Item-Item relations** include categories or ontological concepts. An example for binary item-item relations are similarity relations for information-retrieval.

1.3.3 Collaborative Information- and Knowledge Spaces and the Web

What makes communities so valuable from the information management perspective, is that their CIKS typically has a much more narrow thematic focus and a smaller extension than e.g. general purpose web-directories. This allows to use more specialized information management applications, e.g. more specialized search heuristics. Furthermore, members of communities of interest are typically experts in their field of interest. That allows for a much greater semantic depth in the information that is managed (communicated) within those communities. Usually, we have a dense social net within the community that allows for e.g. judging the quality of information and that allows for other types, qualities and privacies of information to be incorporated in the CIKS. It can therefore be expected that information needs that at the present time have to be satisfied with the help of search engines like Google or web-directories like Yahoo will the future be partially satisfied with the help of a set of communities that the user is part of.

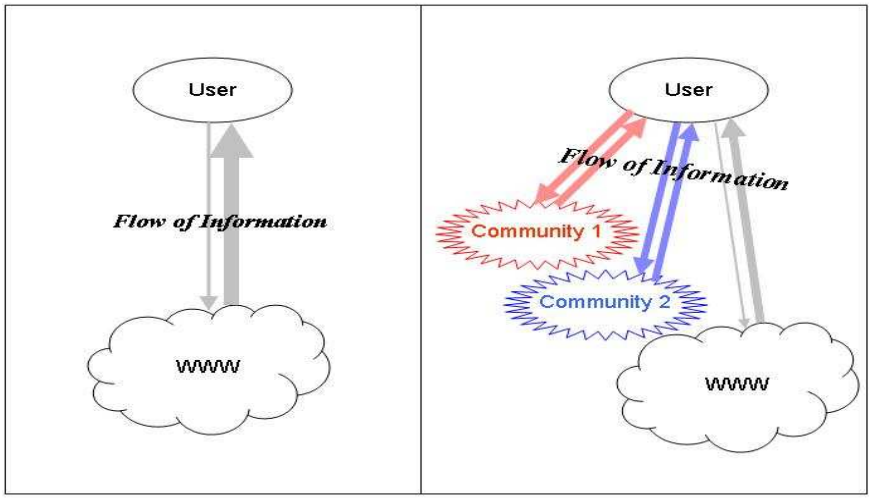


Figure 1.2: (Figure taken from [51]). Qualitative views on information flows to and from the web.

The left part of figure 1.2 shows the "traditional" qualitative view on information flows to and from the web: A large amount of information is transferred from the web to the users. Management, especially retrieval of information is accomplished through large, unspecialized search engines or web-directories. The right part shows the view with communities. For the user, the web does not exclusively appear as a large diffuse cloud (although the view of the right figure is still a valid alternative), but rather as the sum of the thematically specialized communities he belongs to. The information flow from the user to the community becomes a significant contribution.

1.4 Mobility and Context

After basic concepts of communities and virtual communities have been introduced, we will now briefly investigate how mobility and context-sensitiveness influence communities. Before mobile communities will be introductorily characterized, we will briefly discuss the paradigm of mobile computing and characterize the concept of context and classes of context-sensitive applications. The aspects of this section will be dealt with in greater depth in later chapters.

With the advent of mobile computing devices such as advanced mobile phones, smart-phones, wearable computers and PDAs together with wireless transmission infrastructure and protocols (e.g. Bluetooth, W-LAN, GSM, GPRS, UMTS) and sensor technology (e.g. GPS receivers), a new paradigm in human computer interaction was created: Mobile Computing. (see [59] for an overview of the technological issues).
There are various overlapping sub-disciplines of mobile computing such as e.g. Ubiquitous Computing, Wearable Computing and Context-Sensitive Computing. Ubiquitous Computing is about transferring computational and networking capabilities into all sorts of electronic devices of our everyday life thus creating an integrated network environment of interacting information agents [52, 26, 190]. Ubiquitous Computing is sometimes also called Pervasive Computing. Some authors reserve the notion Pervasive Computing to the aforementioned concept and define Ubiquitous Computing as the concept of an ubiquitous access to computing and especially network ressources which allow to use those ressources at any place in the required way. Wearable Computing aims at integrating the computing infrastructure with the human body with the help of Augmented Reality displays, advanced input devices and other concepts [52, 160]. Context Sensitive Computing will be discussed in more depth below.
Mobile computing is different from traditional desktop-interaction because it is much more organically linked with everyday life. Transactions like payment or other financial transactions, exchange of information such as documents or business cards, and access to distributed information like traffic messages, schedules or best price information for some product can be accessed in any real life situation that requires the particular information or transaction. The user does not have to use a fixed terminal but can interact "anytime anyplace" with the services that offer the desired transaction or information. But integrating computing technology ever deeper into human life also implies increased requirements for these services in terms of unobtrusiveness and ease of use, security and reliability, speed and many other aspects. The more a user relies on his mobile computing infrastructure, the lower his degree of tolerance towards malfunctions and bad performance will be. In order to design the human-mobile interface with optimum efficiency, it is necessary to sense and model as much contextual information as possible. E.g. when accessing traffic messages, it is not very comfortable to enter the current location manually into the system, especially when driving. The next section is devoted to characterizing the concept of context in more detail.

1.4.1 Context

In the literature, many definitions for context can be found. Dey [30] defines context as "any information relevant to the interaction between users, their devices and their environments". Schilit [52, 26, 166] divides context into computing context (network connectivity, nearby resources, etc.), user context (location, social relations, etc.) and physical context (temperature, lighting, etc.). This definition focuses on the entities themselves and not on the information

about the entities. Other authors emphasize the difference between context information obtained through implicit vs. explicit human computer interaction [167]. Most definitions of context [30, 57] distinguish between four important categories of context information corresponding to "where" "who" "when" and "what":

- **Location**. This includes virtual locations (e.g. coded as URIs) as well as physical locations. Also includes histories of locations.
- **Identity**. Information and histories of information about the identity of a person or a set of persons which are related to a context-situation.
- **Time**. Information about time of events that relate to a context-situation.
- **Environment or Activity**. Information and histories of information about physical or virtual environments and activities related to a context-situation.

Recalling our general structural model (as given in section 1.3.2) for a CIKS, these categories or classification axes for context are reflected in the following way. Physical location information is part of a person's profile. Virtual location information such as the URI of a document that is currently read by a user is part of the meta data of an item and can be reflected in a relation (in case of the example a user-item-relation). Identity information is also part of the user profile. Time is implicitly involved when histories of context-information are regarded. These histories are either part of a profile, item or relation or are represented as distinct entities. Physical and virtual environment information are mainly included in a person's profile and to a lesser extend in item and relation information.

For the purpose of supporting communities that use mobile computing as a communication means, we can define context in the following way:

> **Context** is the set of explicit and implicit representations of dynamic information in a community's collaborative information or knowledge space (personal profiles, items and relations) that characterize a physical or virtual situation that members of the community are in. Dynamic information is information that changes rapidly over time. A physical situation is an "interval" in time and space and a virtual situation is a time-interval during which a user perceives himself to be in a virtual "place" on the Web. Applications that support a community in the communicative and collaborative build-up of its information and knowledge space must be able to improve their performance for the user in a given situation when using this context information compared with situations when they do not use this context information.

1.4.2 Sensing Context

Context information must be detected and represented in a way that applications can use them. In order to detect context information, many techniques have been proposed [52] which cannot be extensively covered here in detail.
The most important context information is location. We will therefore exemplarily explore the possibilities for detecting and representing locations in more detail.
Location information can be collected with the help of satellite communication e.g. via a GPS receiver. Under ideal conditions, a commercially available GPS receiver can detect its current

position with an accuracy of up to 5 meters [59]. In high density urban areas, accuracy is substantially decreased due to reflections and satellite visibility. Within buildings, GPS cannot be used.
Location can also be detected with the help of cell-based wireless communication like e.g. GSM [129, 88]. Here the stationary broadcasting infrastructure imposes a cell-partition on the service-area (compare figure 1.3). The mobile devices register themselves with the cell that their current location is in and perform a registration hand-over to another cell if the current location changes accordingly. Thus, the mobile device is usually always registered with the cell whose stationary broadcasting infrastructure is nearest. This registration information can be accessed in the mobile device or via the servers of the provider (e.g. O_2). The accuracy of cell based localization

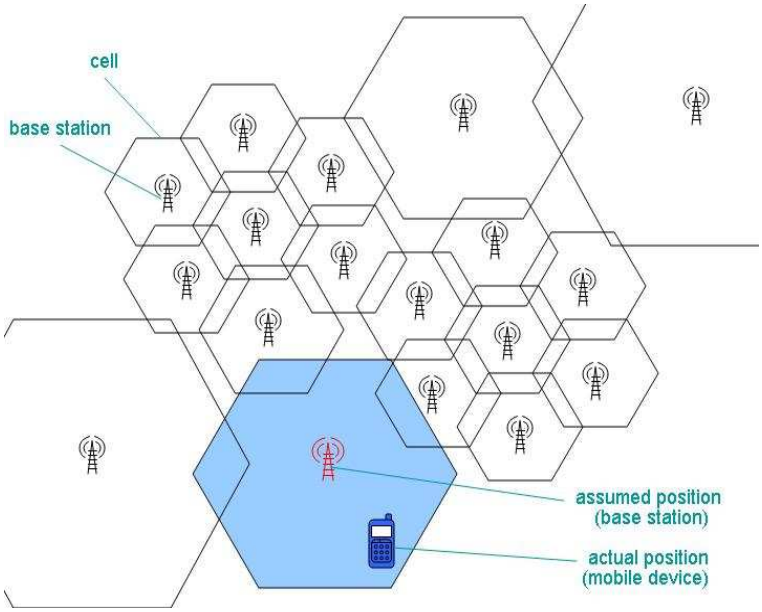


Figure 1.3: Using GSM cells for localization.

is limited by the size of the cells which changes from about $(200\text{m})^2$ to $(400\text{m})^2$ in dense urban areas in western Europe to about $(1\text{ km})^2$ to $(30\text{ km})^2$ in rural areas [88, 59]. This accuracy can be improved by a factor of 2 with the help of triangulation calculations if the location of the neighbor cell base stations and their approximate distance is known.
Another possibility, especially for indoor location is the use of semi-stationary beacons or W-LAN access points. Experiments with active badges or beacons have been conducted at PARC and Olivetti in the late 1980s (see [52]). The basic idea is to attach infrared or radio emitting beacons to interesting locations which can transmit their coordinates and identity to a mobile or wearable device. A more modern approach which is roughly comparable to cell based location is to use W-LAN access points with a known location to locate devices which use this access point.
In order to represent the measured location data in a uniform way, several formats have been developed. Locations which are obtained by cell-based localization are usually given in a complicated geo-format (such as Gauss-Krüger [33]) depending on the telecommunications provider and the region of the earth that one is interested in. The complexity of these geo-coordinate-systems arises because of the fact that the surface of the earth is not a perfect sphere but a

deformed ellipsoid. Different regions of the earth require different fitting surfaces and a transformation of some section of an elliptic surface into a planar map requires suitable projections. More details about representations, models and detection techniques for locations and velocity data will be given in chapter 3. Aspects of other context parameters are discussed in [52].

1.4.3 Using Context: Context-Aware Computing

Context-aware applications make use of a person's context to improve the service(s) that this application implements for that person. As has been stated in section 1.3.2 one can distinguish between personalized applications (making use of rather static profile information), context-sensitive applications in the narrower sense (making use of rather dynamic profile information and relations) and social-sensitive applications (making use of user-user-relations). It will be stated explicitly whether we will use context-aware in the more general or in the more special sense if necessary.

We will now shortly investigate some general fields of applications, which make use of context information. For a more detailed survey consider [30, 52, 26].

Information Access Information needs change according to the context a user is in. It is therefore interesting to provide information proactively (push) or on demand (pull) that matches the information needs in a specific context. The most prominent class of applications that attempt this are tour guides (see [52, 26] for an overview). They mostly make use of location information and information about directions of view to provide information about points of interest (POIs) in a city or a museum. The location of the POIs is matched with the user's location context and information from a database is presented in an appropriate way. Another class are personal information retrieval agents. These systems try to organize a person's personal information space with the help of contextual keys like e.g. organizing documents according to the places they were last needed. Proactive variants enrich the mere time-dependence of a usual organizer by broader context representations and act as "memory augmentation systems" [98, 99, 100, 52].

Communication and Information Exchange Reachability Management applications use context information to adapt communication channels to the contexts of sender and recipient of a communication service e.g. in order not to disturb someone in an important meeting. Furthermore, communication services can use context to offer new variants of communication such as virtually tagging contexts (e.g. places, places at certain times, places at certain times under certain weather conditions etc.) with a message. Automated Information Exchange Applications collect Information in certain contexts like virtual business cards from people in certain contexts (fairs, meetings, etc.) or special offers from shops that match a person's current interest. Voting and payment applications can also incorporate context like identity information.

Computer Supported Collaboration CSCW applications can as well profit from context information. Application fields include collaborative learning environments which can use micro-context information such as the viewing direction of learners to e.g. change the presentation form of learning content. The set of context aware CSCW applications also includes classical group ware applications where e.g. the presentation of shared working artifacts can be automatically adapted to the team's current context (locations, current working status, current working documents). Furthermore, appropriate indications of the contexts of team members as such can aid substantially in the collaboration process.

Automatic Contextual Reconfiguration This field of applications was first suggested in [166]. Reconfiguration according to context can be used to adapt the Locale of an application to the current location or the language preferences of people nearby (especially in collaborative settings). The resource network that an application uses can be adapted to the context: a print command can be directed to the nearest printer or environment parameters like lighting or car seat settings change according to the persons present or according to the time of the day.

Context Triggered Actions Context triggered actions correspond to if-then-type sets of predefined rules that invoke certain actions according to given contexts [166]. This includes emergency situations where certain body parameters trigger a call for help and many more.

1.4.4 Mobile Communities

In the previous sections, a clear distinction has been made between real-world communities and virtual communities. Although mixtures between these two forms are in principle also possible under a mere desktop-interaction paradigm, mobile computing is the main factor in giving rise to mixed real-world-virtual communities. These mixed communities combine and smoothly integrate real world interaction and virtual interaction. Since mobile computing plays a key role in this integration, mixed real-world-virtual communities that use mobile computing as a key element of communication / CIKS-build-up will be called Mobile Communities from now on.
The following observations can be made in terms of mobile computing and its influence on communities in contrast to "normal" virtual communities interacting via desktop computers:

- **Desktop computing** is in most cases characterized by an **isolated interaction context**. One person interacts with a computer and this interaction context usually does not involve physical interaction or physical communication with other people. The user is usually rather isolated from any real-world influences. This isolation has even become a stereotype in social perception since the early 1980's.
- **Mobile Computing** interaction is intended to be **seamlessly integrated** into people's real-world lives. In contrast to a desktop computing session, which usually represents a context of its own, mobile computing aims at supporting a user in already existing real-world contexts. This property is usually expressed in the phrase "anytime, anyplace interaction". Therefore, mobile computing is naturally linked with and complemented by context-aware computing.
- **Mobile Computing** can be used in real-world-contexts which involve **several people** whereas social interaction with groups of people via desktop computers is only possible virtually. As an example for using mobile computing in a social setting consider an application that supports collaboratively choosing a suitable discotheque-event via a smart-phone in the situation of already being out with a group of friends (and not while planning the evening in front of the PC).

It can therefore be concluded that mobile computing is an ideal means for the support of mixed real-world virtual communities, because it links the real-world interaction with the virtual interaction by extending the community's CIKS by real world contexts. This enrichment of the CIKS leads to new requirements in terms of the optimum support for such communities:

- Applications need to be as **context-aware** as possible. It would be awkward for the users in the aforementioned scenario of event finding, to input current location and music preferences.
- User interfaces need to be adapted to allow for a **quick** interaction with the community's CIKS because time is a critical factor in a mobile interaction scenario and poor performance is not well tolerated.
- It is ideal if applications allow for a **collaborative access** to the community's CIKS, e.g. in the upper example showing event alternatives on the mobile devices of all members of the group.

We will come back to these and other requirements and characterizing aspects of mobile communities throughout the thesis.

1.5 An Example: The COSMOS Project

In this section, a mobile community project will be presented. As an example we will introduce in greater depth a community that has been built up in the context of this research project as an illustration of the concepts from the previous sections.

The COSMOS Project ("Community Online Services and Mobile Solutions") [54, 8] is a joint project of the Munich University of Technology departments of Computer Science and Business Management and several partners from industry among which O_2 Germany is the most prominent. Its goal is to investigate properties of mobile communities and requirements for their support. The scientific methodology used in this project is based on three partly overlapping phases. In the first phase of the project, existing communities, mobile communities and technologies were investigated. This is the analysis phase and the phase of inductive, empiric generation of theses concerning the support of mobile communities. In the second phase, three prototype community platforms were created that implement mobile community support concepts which were developed in the first phase. Three pilot communities were established which use the platforms. This phase uses a constructivist methodology with cycles of empirical research and according prototype refinement. The third phase will inductively condense the insights from the piloting phase into general guidelines for the technological and conceptual support, maintenance and build up of mobile communities and community platforms.

There are two distinct piloting fields. The first field deals with the support for communities of cancer patients. In this field, mobility has special aspects such as questions of how the stadium of the disease afects the patient's physical ability to use conventional computer systems and how it correlates with their information- and communication-needs. This results in investigating aspects of micro-mobility (Tablet PCs and PDAs), privacy and community management. The two pilot communities support breast cancer patients and leukemia patients. We will not deal with this piloting field in greater depth in this thesis (see [54] for more information).
The second piloting field is situated in the domain of Lifestyle and spare time activities. It's pilot community is partly recruited from students from Munich University of Technology and partly recruited from a community called "jetzt.de" which arose from a youth- and lifestyle-magazine which was an add-on to a large German newspaper. This piloting community is called "studiosity.de". The CIKS of this community is focused on spare time events from the Munich area. This community will now be investigated in terms of the notions of the previous sections.

1.5.1 Studiosity: Analysis and Architecture

Mobile Communities with a Lifestyle bias were analyzed from the business perspective [8, 154, 155] and from the technological and conceptual perspective [8, 88, 89, 90]. The technological and conceptual analysis was iteratively accompanied by small prototype experiments and small user studies and resulted in the design and implementation of a community platform and the build up of a small test-community. Among the results were the following findings:

- Communities cannot easily be built from scratch. Although several versions of prototype platforms in the Lifestyle domain were promoted in the university domain, none of them reached a stage which would yield substantial dynamics in the community's CIKS. Therefore it was decided that instead of building up a completely independent community it was a better idea to support an existing community ("jetzt.de") with mobile services in order to answer the project's research questions.
- Experiments with WAP and mobile phones as protocol stack and hardware for mobile community support applications did not show encouraging results. This is due to considerable loading time of WML pages and the tedious user interface especially the poor display quality of mobile phones of the years 2001 and 2002. It was then decided to switch to smart-phones and applications using adapted HTML pages for the mobile platform.
- Community support services only function as a tightly integrated bundle. If each service is rather isolated from other services or works only on very specialized parts of the CIKS, it cannot provide support that is tailored for the specific needs of the community. E.g. a reachability management service works quite good, when it integrates profile information (personalization) and works even better when it also includes information about user-user-relations within the community (social-sensitiveness).
- Existing commercial community software (of the year 2001) based on Web-Applications is only restrictedly applicable as a platform for communities and especially for mobile communities. It was therefore chosen to build the platform on the basis of a community framework created at the chair of Applied Computer Science and Cooperative Systems called CoBricks (Bricks for Community Support) [87] with a commercial community application server software (Cassiopeia) [128] as front-end.

These results and experiences from previous community projects [87] led to a concept for a platform architecture for the support of mobile communities see [53]. The concept is based on a model for a CIKS which is similar to the one presented in section 1.3.2. It is typical for a community to have its community platform (the CIKS and the bundle of communication services that operate on it) attached to a Web-Location. The commercial community application server Cassiopeia that is used as a front-end for the platform is a Servlet based Java Web Application Server which is based on straightforward technology for Web-Applications. It contains a number of application components which implement basic community support services like discussion boards, chat, buddylists, session and authentication management etc. and basic CIKS structures like user profiles etc.. The services and the community management functions can be accessed via XHTML websites which contain small XML query- or command-segments which are evaluated by the server and passed to the application layer. The components of the application layer answer the queries with small XML documents or execute the commands. The result of the

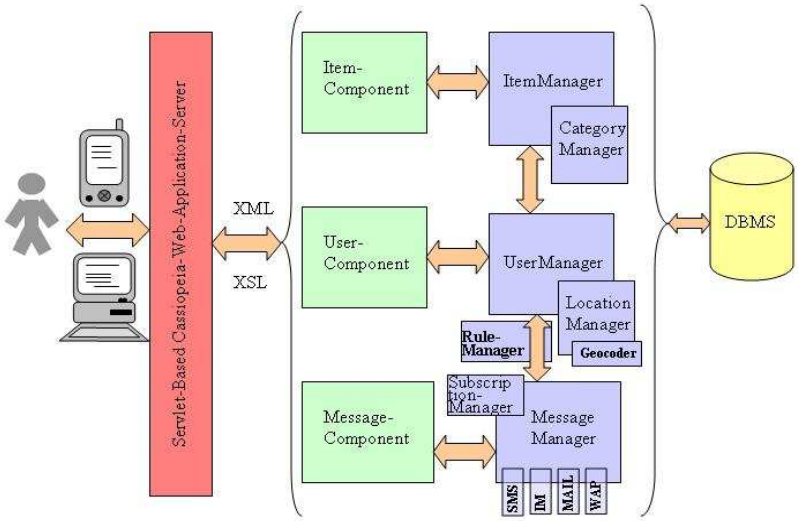


Figure 1.4: COSMOS Studiosity platform software architecture.

queries is then XSLT-transformed back to XHTML, integrated into the XHTML template and served to the user.

Because it turned out that the components were not very well integrated and not easily adaptable to the requirements for mobile, context-sensitive community support, the key components were all replaced by custom components which link the Cassiopeia front-end with the CoBricks backend. Figure 1.4 shows the basic elements of the platform architecture.

The Cassiopeia left-overs (Web-Application Server Servlet, XML/XSLT-processing, Component-Runtime, authentication and few support-components) are depicted in red. The custom components that connect the Cassiopeia front-end with the CoBricks backend are depicted in green and the CoBricks Manager-Modules have a violet color. The main modules are responsible for user profile management, item management and message management.

User Profile Management (User Manager) provides structures for the elements of user profiles and for the access and comparison of user profiles. Elements of user profiles include dynamic attributes such as location and less dynamic attributes such as personal data like name, telephone-number and interests. It also contains the user's personal buddylists. Location Management is associated with the management of profiles. Location data are received either via SMS from the mobile device or via interaction with a service provider server interface. A complete geo-coding system with a detailed set of geospatial data (maps, points of interest, street names etc.) is used to translate the encoded locations into human readable format (see also chapter 3 for further aspects of location data).
Privacy is controlled by lists of declarative rules. These lists of rules are in principle nested if-then-else-expressions. A single rule is composed of a condition part and a return part. The condition part is composed of basic logic operators (AND, OR, NOT) and Java method calls that implement predicates over the attributes of personal profiles of the users. One stack of rules controls the access to a single profile attribute. If a user (or a module in behalf of a user)

asks for a profile attribute of another user, the stack of rules for this attribute is evaluated. The return function result of the first rule whose condition matches is returned. The return values are functions of profile attributes of the asking user and the asked user. As an example, one can consider the 2 element stack (pseudo-code syntax)

$$\begin{array}{rcl} \textit{isEqualWithinRadius(locationOfAsker, LocationOfAsked, 100)} & \rightarrow & \textit{identity(LocationOfAsked)} \\ \textit{else} & \rightarrow & \textit{partOfCity(LocationOfAsked)} \end{array}$$

for the control of a user's location attribute. Any user that is in a sufficient proximity is given the full location whereas other users are only given a coarse grained location information which is reduced to the part of the city.

Item-Management provides structures for managing information items and their meta-information (author, expiration-date, etc.). Meta-Information can include threads of comments to an item. Items can be classified into categories. A special form of category are so called shared buddylists. These lists can be initiated by any user and any user on a list can add new members or delete himself from the list. Items can be categorized to such a list to provide a special form of collaborative access control or control of (semi-)indirect communication recipients respectively.

Message-Management provides support for all kinds of direct communication and reachability management. Messages can be sent to single users, groups of users, buddylists or shared buddylists via several channels (SMS, E-Mail, Instant-Messaging, etc.). The channels preferences of the sender are matched with the channel preferences of the recipient to implement a basic reachability management. A special kind of message are conditional messages which can be put into a queue for a configurable time interval. A condition expression can be given which is composed of basic logic operators (AND, OR, NOT) and Java method calls that implement predicates over the attributes of personal profiles of the users. The conditions are evaluated by the rule processor and lists of recipients that match the condition are returned.
Examples for such conditions are (pseudo-code syntax)

- *isEqualWithinRadius(locationOfPossibleRecipient, LocationX, 100).*
  This condition corresponds to tagging a place ("LocationX") with a message. The message is delivered to all users that enter a circle with a radius of 100 meters around that location in the separately specified time interval.

- *hasSimilarityOf(interestsOfPossibleRecipient, interestsOfSender, 0.5).*
  This condition corresponds to sending the message to all users that have personal interests that are at least 50% similar to the interests of the sender.

- *isEqualWithinRadius(locationOfPossibleRecipient, LocationX, 100) AND hasSimilarityOf(interestsOfPossibleRecipient, interestsOfSender, 0.5).*
  A combination of the previous two conditions. The message that has been tacked to the specific location will only be delivered to those people that enter the surroundings of that location and have similar interests.

See [53, 87, 88] for a more detailed discussion on the architecture of this platform and the background of the CoBricks system.

According to the CIKS model in section 1.3.2 and to the discussion on community support requirements in section 1.3.1, the architecture from above is only a first basic step. Improvements

in terms of service integration and declarative modeling of the CIKS are highly desirable. The user profiles and information items roughly correspond to the profiles and items from section 1.3.2. What is not explicitly represented in the upper architecture are relation objects. User-User-Relations are implicitly incorporated in the user-profile management (personal buddylists) and in the item/category-management (shared buddylists). User-Item-Relations are also only implicitly present, e.g. in the author meta information of an item. The item-management implements parts of the indirect communication requirements of section 1.3.1. and the message management implements parts of the corresponding direct and indirect communication requirements.

1.5.2 Studiosity: Web Platform and Mobile Platform

The user-interface of the architecture has been implemented as two sub-platforms, web-platform and mobile platform. Screenshots of web-platform are depicted in figures 1.5 and 1.6, screenshots of the mobile platform are depicted in figures 1.7 and 1.8. Because the associated community was more or less restricted to the Munich Area, the platform uses German language.

The **web-platform** page structure is coarsely divided in 3 sections, a more status-oriented/functional toolbar on the left edge, a more thematically oriented toolbar on the upper edge below the graphic, and a content pane with white background in the center of the page. The original page design was subject to several changes which gradually blurred the distinction between the originally more functional role of left toolbar and the originally more thematic design of the upper toolbar which was shrunk to the fields “Freunde” (friends), “Events” and “Infopoint”. The friend area is dedicated to the management of buddylists and shared buddylists. New lists can be created there and profile parameters of the buddies can be accessed. These include online status and locations which can be displayed on a map. The Events section contains information item management where each item corresponds to an event. Comments can be added and events are categorized into several fixed categories and can also be restricted for shared buddylists. An event stream from a local Munich event agency is fed into the database to provide an initial set of events. Infopoint points to several pages which contain fixed information like e.g. event-related locations etc..
The screenshot in the upper left corner of figure 1.5 shows the index page. The left toolbar contains a login area, information about the number of users currently registered and access to a chat and some discussion boards. The lower left picture shows the MyStudiosity page which for a logged-in user allows access to personal data management functions like management of personal profile information. Additionally, the left toolbar contains a linked overview of the message management and an overview of the login status of other users with associated communication channel links. The upper right picture shows the interface for the management of the personal profile. Every profile parameter can be assigned one of 3 rule stacks which correspond to high, medium or low privacy (corresponding to no access, access restricted to buddies and shared buddies, and access for all members). The last screenshot shows a separate menu for controlling location data. Locations can be predefined and chosen in case the automatic location is not available or not desired. Privacy settings for location can be chosen in a more fine grained manner, allowing not only to restrict the set of persons that can access the location but also allowing for the configuration what these sets of persons can see from the precise location (city only, city and part of the city, full access).
The screenshots in the upper row of figure 1.6 show the start-pages of the friend area and the

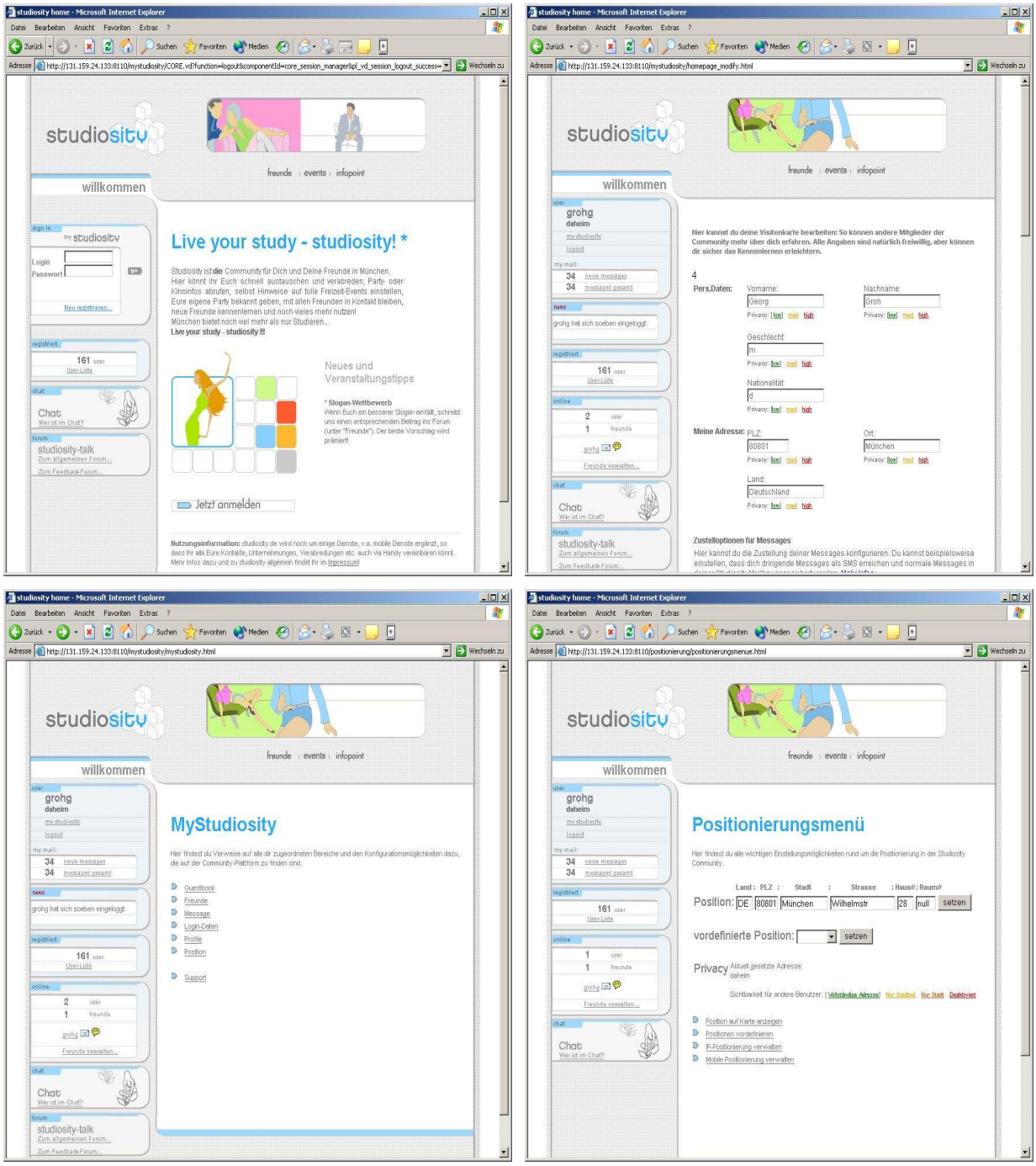


Figure 1.5: COSMOS Studiosity platform screenshots, Part (I).

event area. The event area contains links to events in fixed categories and a link to events for the shared buddylists that the logged-in user is member of. The lower left picture shows the category "parties" with one event. Detailed information and the comment threads can be accessed from this page. The screenshot in the lower right corner shows the message manager main page. From the vast possibilities for conditional messages that have been discussed in the

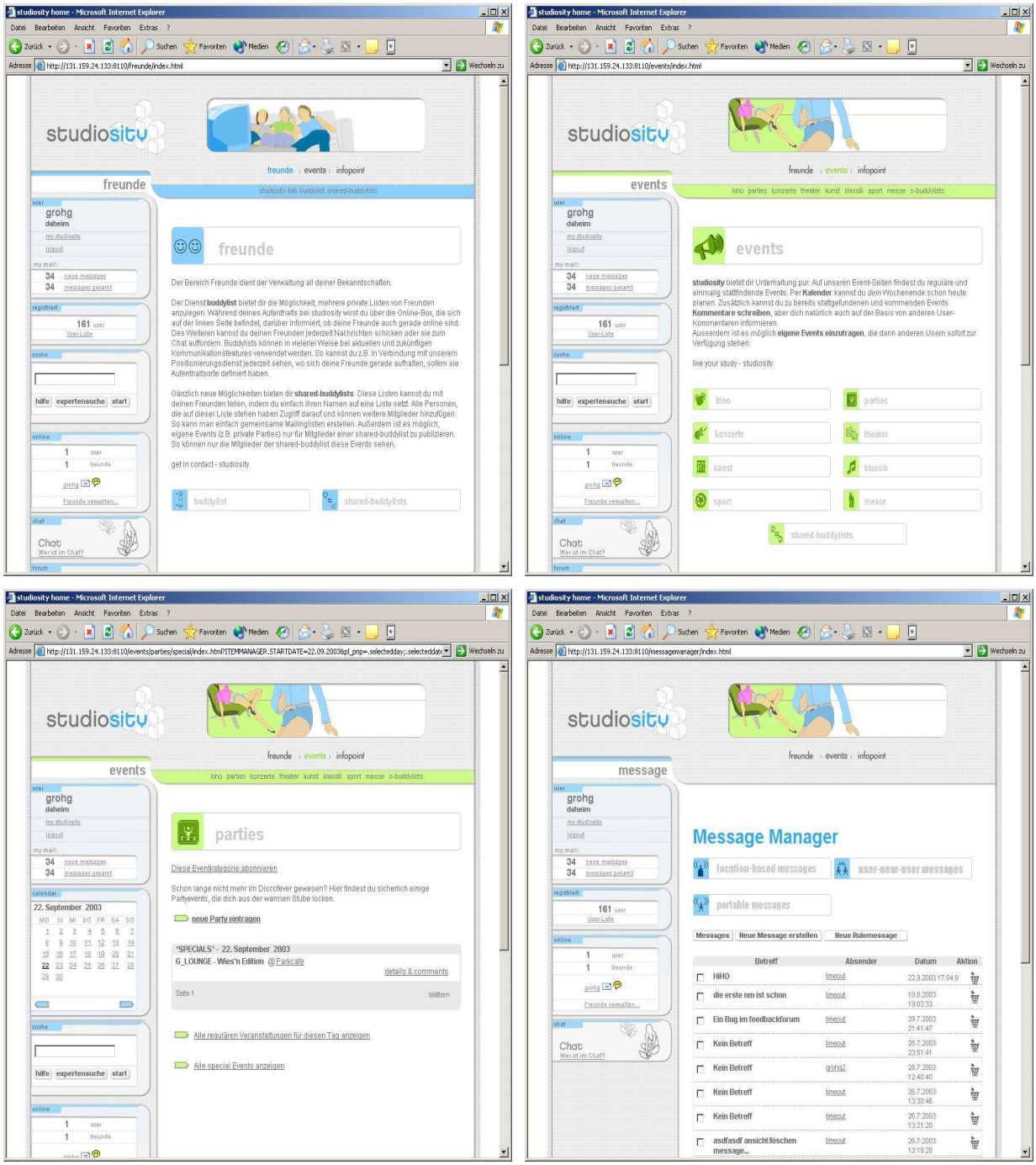


Figure 1.6: COSMOS Studiosity platform screenshots, Part (II).

previous section, three forms of messages have been implemented in a convenient way:

- **Location Based Messages** are messages which can be tacked to a fixed location. The user can specify whether only special users or users from buddylists or shared buddylists will receive the message or whether all users that enter the location will receive the message.

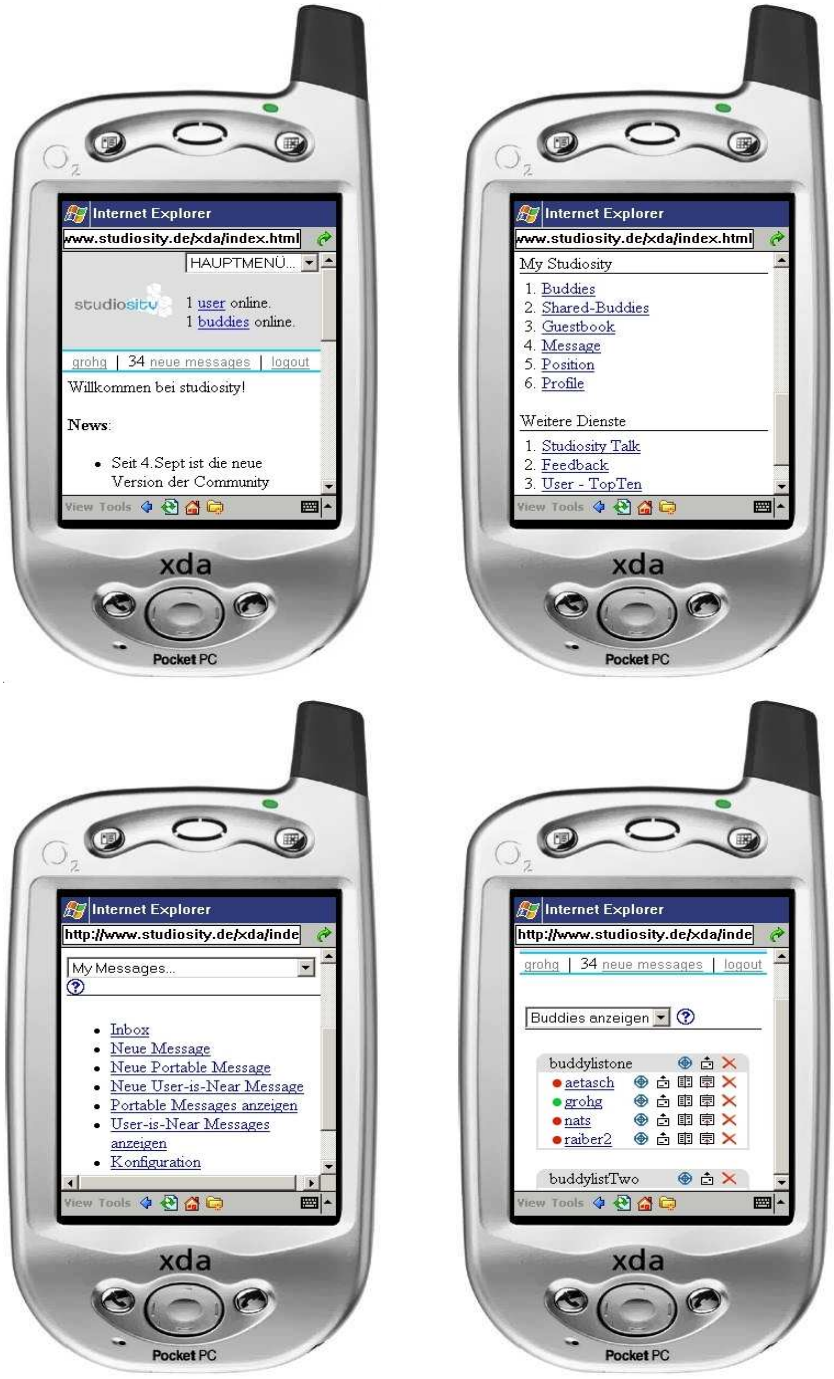


Figure 1.7: COSMOS Studiosity mobile platform screenshots, Part (I).

- **Portable Messages** are messages which a user can virtually carry around with him. Other users that he approaches will get the message. The set of recipients can be restricted in the same way as with location based messages.
- **User near User Messages** represent the opposite concept: If somebody enters a user's

Figure 1.8: COSMOS Studiosity mobile platform screenshots, Part (II).

location, the user will be informed with a message.

Furthermore, users can build their own condition expressions giving unlimited access to the full spectrum of possibilities described in the previous section.

The **mobile platform** is based on the hardware side on a smart-phone from O_2 called XDA. It

combines a multi-band GSM / GPRS cell phone with a Windows PocketPc based PDA. With GPRS, a volume based pay scale with a permanent Internet connection can be realized. The user interface consists of adapted HTML pages. Figures 1.7 and 1.8 show screenshots which depict the device with its 240 x 320 pixel touch sensitive color display which is used with a pen and the PocketPc version of Microsoft's Internet Explorer showing the pages.
The screenshot in the upper left corner of figure 1.7 shows the welcome page. In the upper left corner, the main menu is accessible from every page via a drop down menu. The number of users and buddies online is shown together with news concerning the platform. Scrolling below yields the screen of the upper right corner, showing the options of the main menu again in link form. The spectrum of services matches the spectrum of services from the web platform in an adapted form. Clicking on the "Messages" Link results in the page in the lower left corner. The various messaging options can be accessed via links. For every menu item of the main menu (here the "Messages" section), a drop down sub menu exists with various configuration and access options. The mobile version of the buddylists page can be seen in the lower right corner. For every buddylist, the iconified quick access options include the option to locate all the members in the buddylist either on a map or via textual display (via the crosshair symbol), to write a message to all members (via the envelope symbol) or to delete the complete list (via the "x" symbol). The same options exist for every member of the buddylist (plus options for member-guestbook entries and options to show the profile ("visiting card") of the person).
Locating all members of a buddylist on a map is shown in the lower left corner of figure 1.8. The individual person is depicted as a circle on the map. In the lower right corner, a screen is shown, where users can manage their position is automatic localization is not available or not desired by choosing from a set of self-defined locations or entering the address manually. The page in the upper right corner shows the reachability management configuration. for every priority degree that senders of messages configures for those messages, a user can chose over which cannels he wants to receive the respective message. Finally, the screenshot in the upper left corner shows the mobile version of the menu for portable messages where a radius for the circular area can be defined which specifies a "location".

Summary

General communities are sets of people distinguished by membership awareness, a common pursuit, and a certain degree of similarity and strong communication relations among its members. Virtual communities use electronic media for communication. All actions in a community can be modeled as communication acts. Computer mediated communication (CMC) is characterized by a strong emphasis on asynchronousness and semi-asynchronousness, indirection, anonymity and alternative identities and text-based form. CMC in virtual communities incorporates aspects of social relations among its members. It has a strong emphasis on n:m cardinality. Communities regarded in this thesis can be characterized as a set of people which have a high degree of community-awareness, communicate with other members via electronic media, and have a common pursuit which can be identified with the pursuit to collaboratively build up a thematically focused, information- or knowledge-space. This collaborative information- or knowledge-space (CIKS) predominantly contains semi-formal implicit "warm" information or knowledge with a strong emphasis on textual form. The concept of a collaborative information- or knowledge space differs from the concept of a knowledge base in artificial intelligence. A community differs from a team in terms of size, goals and working objects. In terms of community support, we have to distinguish between the community, its data-representation, its theoretical model, algorithms or methods for community support, implementations of these, and a concrete instance of these implementations together with the data representation associated with a particular community (a community platform). Many commercial software packages for community support already exist. Requirements for community support are equivalent to requirements for the managements of its CIKS, or equally equivalent to requirements for direct and indirect communication. Direct communication support requires channels for all cardinalities, conditions, levels of synchronicity etc. together with a reachability concept. Indirect communication support requires means to collaboratively change, add or delete information or knowledge in form of communication content. It also requires means for the management of content-user relations and of user-user relations. In a community, social awareness and fine tuned integration of all support services is essential. A CIKS can be modeled as a set of information- or knowledge-items, user profiles and relations between items, between users and between users and items. Communities and their CIKSs can act as an informational buffer layer between a user and the web. The paradigm of mobile computing is different from traditional desktop-interaction because it is much more organically linked with everyday life ("anytime, anyplace" interaction). Deep integration into the everyday life and altered performance-, fault- and ease-of-use-tolerance levels require more context-sensitiveness from the services. For our purposes, context can be characterized as the set of explicit and implicit representations of information in a community's collaborative information or knowledge space that characterize a physical or virtual situation that members of the community are in. Context awareness must improve the usefulness of an application. Typical general application fields for context sensitiveness are information access, communication, CSCW, context triggered actions, and automated contextual reconfiguration. Mobile communities are mixed virtual-/real-world-communities using mobile computing. Mobile computing in communities is characterized by a social interaction context. The COSMOS Lifestyle project together with its Studiosity.de community is an attempt to empirically and prototypically investigate mobile communities.

Chapter 2

Groups and Ad-Hoc-Groups

In this chapter we will review the notion of a group from the point of view of other scientific disciplines especially from the point of view of social-psychology and mathematical psychology. We will work out basic characteristics of human groups and will discuss how the formation of groups interrelates with common context parameters such as location. We will review Social Identity Theory and other special theories that characterize normative effects of groups. Conclusions from these theories and the previous considerations will be part of the foundation for later Ad-Hoc-Group and abstract group analysis and applications. We will then discuss graph-theoretic approaches from Sociometry and introduce formal methods for group definition and detection and explain the relation to hierarchical clustering methods. The chapter is concluded by a summarization and conclusion from the previous discussion that is intended as an agenda and justification for the following chapters.

2.1 The Notion of a Group in Sciences

In the previous chapter we have reviewed some basic facts about communities and mobile communities. The discussed definition frame for a community has emerged and has been influenced from contributions from a wide variety of scientific disciplines such as sociology and computer science, as has been discussed in chapter 1. This definition frame or consensual characterization emerged in parallel to the actual development of virtual communities on the web and led to the accepted characterizations of virtual communities which perceive virtual communities as special forms of (virtual) social groups with specific characteristics.
The emergence of mobile communities broadens and softens this new definition frame again by introducing context sensitiveness and a **deeper embedding** of community services into the user's everyday life. In such **mixed real-world-virtual mobile communities**, social interaction patterns which also exist in the **real world** and are not based on virtual interaction alone, are supported by community services. Purely virtual communities are in most cases "bound" to a single community platform which requires a unique user identification and thus technically defines the extension of this purely virtual community. Furthermore, the users of such a purely virtual community are usually "confined" to the services and communication channels for interaction because no real world connections are available to them. In contrast to that, mobile interaction in mixed real-world-virtual mobile communities not just introduces one more communication channel but smears out ("broadens and softens") the boundaries of such communities because the deep embedding of access to mobile community service into real life

adds the whole spectrum of real world interaction means (face-to-face, telephony etc.) to the community's interaction possibilities. As an example, imagine a group of friends which go out clubbing. Two of them have access to a mobile community platform while two others have not. Another one of the friends is on his way to join the other four and is in cell-phone and SMS contact with the others in order to coordinate their meeting point and later evening activities. Although technically these two plus one friends are not registered to the mobile community platform, they may as well indirectly access the community's services by real world interaction with the two registered friends. This could include the request to look up club recommendations in the mobile community's CIKS as well as the request to contact another mobile community member through one of the mobile community's communication channels. In that way the mobile interaction paradigm softens and broadens the boundaries of "classic" virtual communities. So, as has been stated above, in such mixed real-world-virtual mobile communities, **groups** which also exist in the **real world** and are not based on virtual interaction alone, are supported by communication technology and community services. Within the framework of a larger community which provides a common ontology, a common platform and an organizational frame for community services, these **mixed-real-world-virtual subgroups** are interesting objects of study. In order to be able to discuss these phenomena on a solid conceptual basis, it is necessary to review what other branches of science have to say about **general characteristics of groups**.

Groups play an important role in almost any branch of science that investigates structures which are either human generated or human related. Such sciences include computer science (e.g. teams in groupware) law science (e.g. groups as legal entities), economics (e.g. working teams), ethnology (ethnic groups), history (e.g. social and political groups of the past), art (e.g. artist groups) etc.. While all these scientific disciplines investigate rather special aspects of groups, sociology and especially social psychology try to characterize groups from a more generic point of view. Since we are ultimately interested in improving and structurally investigating support services for such groups and especially sub-groups that fit our rather general definition of a community or sub group of a community, we will not make specialized assumptions about the concrete focus or common pursuit of these groups. Thus we will have to rely on the scientific discipline with the **most generic point of view** as a starting point.

2.1.1 The Notion of a Group in Social Psychology

The field of research in sociology and **social psychology** which deals with groups like we have them in mind is usually designated **small group research**. The term small group attempts to distinguish the scientific subject from sets of people of the size of political parties, ethnic groups and the like. The size of the investigated groups will be discussed below in more detail.

Early contributions in small group research reach back to 1926 (see [13]). With a peak activity in the 1950's and 1960's the field regained attention in relation to virtual teams and communities in the late 1990's [13]. For a brief summary of the notion of a group see e.g. [65, 9, 36]. A more detailed discussion can be found in e.g. [40] and [63, 72] provide a deeper presentation of small group research.

Clearly, the **notion of a group** is a very general one when not specifying further characterizing features. Nevertheless, there is some agreement with respect to a basic definition or characterization of the term in social psychology. While **Individualist** school of thought emphasizes that all phenomena in social groups can be modeled, investigated and derived by investigating the dyadic relation between the group members [40, 69], **Collectivistic** school of thought assigns a

reality and characteristic parameters to a group independent of its members. The more modern approach is an individualistic approach which emphasizes **Emergence** and emergent features of a group which are not directly apparent by analyzing individual members. The phenomenon of emergence (see [178]) is a system theoretic concept which emphasizes that a system which is composed of simple subsystems can show a complex behavior or has properties which are only indirectly coded in the simpler subsystems and cannot easily be deduced from the behavior / properties of the subsystems (e.g. a beehive).
Emergent modeling and collectivistic view do not differ in effect very much, because they both predict properties for the group which cannot easily be deduced from the properties of the dyadic relations between group members.
The **main characterizing features of a group** are described in the following points:

- The **minimal definition** of a group is a comparatively small number of people which interact with each other directly via face-to-face-interactions [72, 40]
- The **number of group members** is usually so small that direct **face-to-face-interactions** are possible between all members [72, 40]. An often stated number for an upper bound is 20 [40]. This may also be justified by considering results from cognitive science which suggest respective limitations of human cognition and perception [175].
- The interaction situations must be of a **certain duration** in order to allow for common structures like norms or goals [40].
- Group members share a **network of interpersonal attraction** (Hare in [13]; [63]).
- Often, the members of a group have interdependent characteristics: **common goals**, **common norms**, a **special communication structure**, a **role- and affect structure**, and a **group awareness** [40, 13].
- Groups are often characterized by **immediately perceivable features** (like names, uniforms etc.) which allow others to perceive the group as a whole and which define borders of the group. [36, 65].

A definition from Homans (1950) [72], which is an often cited common denominator in small group research sums up the notion of a group:

> "A group is a number of persons who communicate with one another often over a span of time, and who are few enough so that each person is able to communicate with all the others, not at second hand, through other people, but face-to-face."

Several aspects characterize special forms of groups. The first aspect is the **size of the group** Social psychology distinguishes between **small groups** and **large groups**. It is usually agreed upon, that with increasing size there is a decrease in the group's "quality" [40] (with respect to group coherence, group self perception etc.) and a subdivision of the group into cliques and subgroups becomes probable [40]. The notion of a (small) group is to be distinguished from the notion of a **quasi group** (sometimes also called statistical group) which denotes just a set of people that have a statistical property in common (such as e.g. color of skin, age etc.). Other terms are **crowd** which denotes a set of people that stay in a common location or **mass** which often denotes an emotionalized crowd [40].

A second categorization of groups is concerned with the social role and social importance of a group for an individual in the group. A **primary group** is a group characterized as a group which has special importance for a group member (a typical example is the family). There are usually only very few such primary groups while a person can be a member of a large number of **secondary group**s which are of less significance. In terms of research in group interaction the notions of **in-group** and **out-group** also play a role. From the view of an individual, the in-group is the own group and the out-group are perceived as "the others" [40, 9].
Also interesting is the notion of a **reference group**, which is an in- or out-group to which emotional or cognitive ties exist [40]. Socio-psychological research results indicate that reference groups are chosen by an individual on the basis of perceived similarity between the person's social personality to the whole group or to some members of the group [40]. Attraction to a reference group is among other factors fostered by the effect of **propinquity** [40]. Propinquity effects are not limited to reference groups. They show themselves in many forms of dyadic and group social relations and will be discussed now in more detail.

2.1.1.1 Propinquity Effects: Social Relevance of Space

For the later discussion in this chapter and for the rest of the thesis, effects of context parameters like location and velocity on social structures and vice versa and especially the usefulness of such contextual parameters for indicating social relations and structures are of special interest. While the last point will be discussed later, we will now take a closer look at the effects of location on social structures.
In classic **CSCW**, several studies have been conducted on how **space influences social aspects** of computer mediated collaboration. As is pointed out in [14], it was e.g. found out in the study [95] that people were more likely to collaborate, the closer they were located within a building. Furthermore, studies like e.g. [151], reviewing research on distributed and collocated work, show that the degree of success of the collaboration can be attributed to factors of distance as is concluded in [14]. In [5] it is stated in the context of discussing technological implications of communication that "if you are farther than **30 meters** away from somebody you might as well be several miles apart" (as cited in [14]).
The study [145] investigated similarity and propinquity effects on friendship formation. In this discussion the term **propinquity** can be defined as nearness of people in time and space. Sometimes it is defined as nearness in time *or* space [14] and sometimes it includes other similarities among the people in question. The study [145] defines several fuzzy regions (radii) of increasing distance such as family space, neighborhood space, economic space and urban regional space that have decreasing social effects. It states that the shorter the (average) distance to a person is and the more similar the person is the less attractive she or he must be in order to perform actions that aim at establishing or maintaining social relations to that person. It investigates the setting of a group of students in a student dormitory to support the theses. The effects of spatial distance as a cost factor or barrier for social attractiveness have been found by a number of other studies as well [40]. Modern social psychology assumes that these effects are not only due to "mere exposure" [40, 101]. The **"mere exposure" theory** by Zajonc [198] states that the mere exposure to a social stimulus increases the attractiveness of this stimulus. In application to location that means that people that are less distant on average and thus have a higher chance of presenting their social stimuli (properties) to a person increase the attractiveness of these properties to the person.
In [14] the **effects of physical distance** on three important social indicators / social phe-

nomena was investigated. People were exposed to experiments were they had to interact with another person through electronic media and were either told that the person is in the same city or that the person is several thousand miles away. First, **cooperation** was measured by exposing persons to a prisoner's dilemma game. Secondly, people were confronted with a standard setting to measure the degree they could be **persuaded** by the other person to change certain views and the last experiment was a standard experiment to measure the degree of **deception** when communicating aspects of self estimations. In all three experiments, significant influences of distance on the person's cooperation, persuasiveness and deception could be confirmed.
A more concrete theory of how propinquity influences social structures is integrated in Latané's **Social Impact theory** [101]. This theory states the social influence on a person (the **social impact**) is a function of three variables (see also [40, 102]):

- the **strength** S of the source (prestige, persuasiveness, etc.).
- the (physical) **immediacy** I (in time and space) of the source.
- the **number** N of sources.

Latané states that the Social Impact SI on a Person is a product of these factors:

$$SI = S \times I \times N^c \quad \text{with} \quad c \in \mathbb{R} \quad . \tag{2.1}$$

He views a person as being exerted to psycho-social "force fields" whose strength is reflected in social impact which closely resembles a field theory in physics.
The theory has been verified in various studies and has been found to be consistent with a large number of social psychological observations [40, 102].
Of special interest to the discussion in this chapter is that he proposes as spatial dependence of I an inverse power of the radius (distance) [101, 102]:

$$I \sim \frac{1}{r^2} \tag{2.2}$$

In [102] this aspect of his theory was investigated by three experiments were the number of memorable interactions of people with other persons in correlation to their average physical distance to that persons was measured in three different social and experimental contexts. Under the assumption of even spatial density distribution of persons, the number of persons $n(r, \Delta r)$ in a circle ring is proportional to

$$n(r, \Delta r) \sim \pi((r + \Delta r)^2 - r^2) \underset{\Delta r \to 0}{\sim} r$$

which was wrongly derived in [102] although the dependence $n \sim r$ was correctly stated.
Thus the number of memorable interactions with persons at a certain average distance r which is assumed to be proportional to the social impact of these persons should be proportional to $r\ 1/r^2 = 1/r$ [102] which was very well confirmed by the study.
Latané studied the group-level consequences in settings of spatially fixed but evenly distributed agents acting on one another over the course of time according to social impact theory [103]. The laws found and the observations made were condensed in his so called **dynamic social impact theory**. Various discrete "social geometries", as he calls them, were tested in computer simulations. Such geometries correspond to 2 dimensional spatial nets or grids, were the grid's

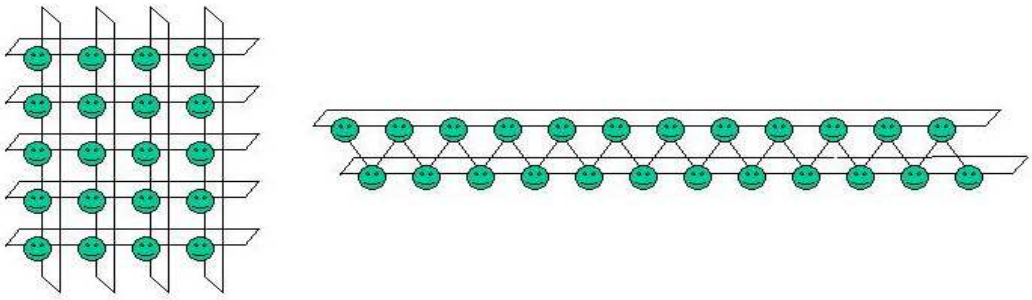


Figure 2.1: ([103]) Social geometries described in [103]. The smilies represent agents and the edges represent communication channels (or more generally possibilities to exert social impact). The left structure is a torus and the right structure represents a "ribbon".

edges indicate possibilities to exert social impact and the nodes correspond to agents (people). Figure 2.1 depicts two examples of such grids were each agent is exposed to the social impact (e.g. through communication) of four other agents. Numerical simulation of social impact theory shows that despite varying the geometry, algorithms and boundary conditions, three main group level phenomena occur which are key parts of dynamic social impact theory: Consolidation, clustering and continuing diversity [103]. **Consolidation** is the effect of reducing the diversity (e.g. in opinions) in a society, **clustering** is the effect of the formation of spatially compact subgroups with homogeneous attributes (e.g. opinions) and **continuing diversity** (incomplete consolidation) is the effect of minorities surviving as a result of being able to resist adverse social influence by the majority [103].
The same effects were found in a study with real persons which took part in an experiment, were they were asked to predict the choice of the majority of their "society" out of two alternatives. They were given message channels to 4 neighbors in various social geometries and an experiment with randomly changing communication channels was conducted as a control means. Although this game is rather simple, the authors found their theories of consolidation, clustering and continuing diversity well supported in the study [103].
What we can generally conclude from the cited studies and from social psychology literature is that space is a major influence on social relations. We will now shortly review a theory which deals with the group influence on a person in terms of this person's social identity.

2.1.1.2 Social Identity Theory: Self-Perception and Groups

Tajfel created a theory of social identity [182, 184] which is according to [14, 40, 182, 184] characterized by the following points:

- People obviously have a tendency towards a positive self estimation. The positive self estimation is often mediated by a **social identity** which is formed through identification with one or more **groups**
- People **subjectively belong to a group** and derive positive or negative evaluation of their own social identity by evaluating the positive or negative standing of this group within the society.

- The own group is compared to other groups (in-group $\leftrightarrow$ out-group)
- People have a tendency to **categorize** others. Members of other groups are perceived to be far more **dissimilar** to oneself than members of the own group.
- People strive towards **positive distinction** of their own group (and as a result of their own social identity substantially formed by means of perceived group memberships) with respect to other groups.

The last point is especially interesting with respect to context, because it is stressed in [40] that positive distinction can be conciliated by being in similar contexts with other people (similar locations etc.) and that vice versa people strive to reach positive distinction by e.g. **segregation** (deliberate (spatial) separation from other groups). Bradner subsumes context influence in social identity theory by stating in [14] that "social identity theory suggests that we are less likely to trust, cooperate and attribute expertise to individuals who are further from us, compared to those who are near".
The main statement of social identity theory is that we define ourselves through the groups we perceive to belong to. This will be of importance in the next sections.

2.1.2 Sociometric Descriptions for Groups

We have seen that groups play an important or even key role in defining our social identity and that context (especially space) has a substantial social influence. We will now shortly investigate, what structural elements of groups are topics in social psychology and then discuss how these structural elements are formalized in sociometrics and mathematical psychology.
In social psychology, several **structural dimensions of groups** are investigated. The first aspect is how group structures generally develop. According to [40], Cartwright and Sanders distinguish between 3 main general influence areas for the emergence of group structures: One area is the structuring imposed by the **group's goal**. This includes e.g. organizational patterns found in professional working teams. A second area is defined through the **personal properties of the group members** (self-confidence, intelligence, qualification etc.) and the third area is about influences from the **environment** (society, super-groups etc.).
Structuring dimensions dealt with in social psychology include objective and subjective group structure, formal and informal structure, affect structure, status and role structures, power and leadership structures and communication structures. Without going into detail about all the various approaches with respect to explaining these structuring directions, it can be stressed that most of the models proposed describe the emergence of group level structures by investigating the net of dyadic social relations between the group members [40, 69] which we will come back to below.
One interesting quantity that can be derived from these formalizations is **group cohesion**. Informally, group cohesion is defined as the average binding strength within a group [40] or "the tendency of group members to stick together" ((Sproul and Kiesler) see [13]) or "in-group favoritism" ((Tajfel) see [13]) and is closely linked with the strive for positive distinction as described in the previous section. An example consequence of high group cohesion is that there is tendency of such groups to develop uniform opinions or points of view which is also consistent with dynamic social impact theory. Festinger [39] conducted some fundamental studies in the 1950s that support the theory that people are in an ongoing process of evaluating their opinions

with respect to the group's opinion and that this continuing process creates a "socially persuasive force that acts to reinforce the majority opinion" [13]. Several subsequent studies have confirmed this since then. This tendency which is also a key part of social identity theory has been called "Groupthink" by Janis in 1977 (see [13]). We will come back to this in the next section.
Group cohesion can e.g. be formalized by the fraction of mutually existing dyadic ties ("mutual choices") compared to the maximum number of possible dyads in a group of n persons as [40] (see also [188])

$$\text{Group-Cohesion} = \frac{\text{Number of mutual ties}}{n(n-1)} \tag{2.3}$$

As has been stated above, emergent group structures in cohesive groups can be investigated by formalizing the **net of dyadic social relations** among group members. (Compare the short discussion about individualistic vs. collectivistic schools of thought and emergence in section 2.1.1 above). **Main factors for cohesion** according to [188] are mutuality of ties, closeness or reachability of group members, frequency of ties between group members and the relative frequency of ties within the group compared to ties between group members and non-members. **Formalizing** these nets of social relations allows for fine grained and mathematically consistent characterizations of groups and group structures which far exceed simple expressions such as the expression for group coherence cited above.
Sociometry is devoted to model the nets or graphs of social relations or ties and to describe emergent structures that manifest themselves in these models. We will now review some basic models from sociometry for cohesive subgroups described in [188].
If we **model human social relations** in a graph $G = (\mathcal{V}, \mathcal{E})$ with persons v_j as vertices (nodes) $v_j \in \mathcal{V}$ we can model human dyadic relations $e_{j_1,j_2} \in \mathcal{E}$ as edges. We have various possibilities to formalize these edges according to the type of relation we want to model and according to the accuracy we intend to model them. The first type of properties of social relations we have to take into account are their algebraic properties. That is whether the relations are symmetric, transitive, anti-symmetric etc.. The symmetry of the relation decides if we need directed or undirected edges in the graph. The second property is whether we assign a strength value to the relations or not. For example we can introduce unlabeled ties like "likes" where we restrict ourselves to either modeling sympathy or dislike (strength 1 or 0) or we can introduce strength labels to the edges (e.g. on an ordinal scale like ("very good", "good", "average", etc.) or on an interval / ratio scale like in $[0, 1]$) which allows for a finer grained description. Let us first look at group models in case of unlabeled, symmetric relations.

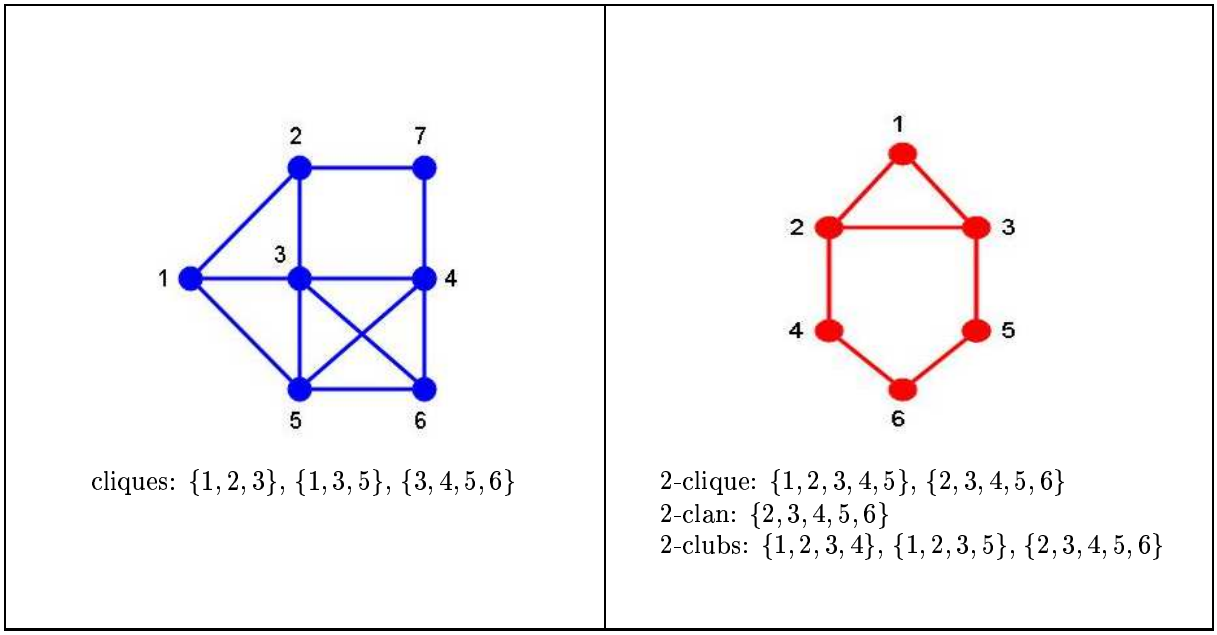
2.1.2.1 Undirected Graphs

The simplest structural definition or model of a group in a society $\mathcal{V}$ is a **clique** g_{clique} which is a maximal complete sub-graph of G with three or more vertices [188]. A **maximal** property (here the property of completeness) of a sub-graph is defined as a property which holds for the sub-graph but does not hold anymore, if vertices from outside the sub-graph are added together with their edges to nodes in the sub-graph. Clearly, this definition is too restricted because the removal of only one edge in a complete sub-graph destroys the property of completeness and thus the clique property. Furthermore, since all vertices are equivalent due to completeness, no substructures in the group like e.g. core actors (people with a higher connectivity) can be described / discovered.
Several approaches for loosening the definition / model for a group without leaving the ground of

well defined mathematics have been proposed [188]. One possible class of alternative definitions use reachability between nodes and diameter of the graph. The reasoning behind that is that people in cohesive groups do not all have to be directly connected. It often suffices for one person to "reach" other persons via intermediaries. One such alternative to cliques are **n-cliques**. A n-clique $g_{\text{n-clique}}$ is defined as a maximal sub-graph where the geodesic distance $\text{dist}(v_{j_1}, v_{j_2})$ (the minimal path length) between any two vertices in the sub-graph is less or equal to n:

$$\forall v_{j_1}, v_{j_2} \in \mathcal{V}_{g_{\text{n-clique}}} : \text{dist}(v_{j_1}, v_{j_2}) \leq n \tag{2.4}$$

Problems with this definition are that the diameter of the resulting sub-graph may be larger than n and that the resulting n-clique may even be disconnected. Both effects are due to the fact that the definition does not exclude nodes from outside the n-clique to be contained in the minimal path of maximal length n connecting two nodes in the n-clique.
The negative aspects of the n-clique definition can be avoided by restricting the paths to nodes within the group or by restricting the diameter of the group to n. The first restriction leads to so called **n-clubs** the second restriction leads to so called **n-clans**. All n-clans are also n-cliques and it can be shown that all n-clans are n-clubs but not all n-clubs are n-clans [188].



cliques: $\{1,2,3\}$, $\{1,3,5\}$, $\{3,4,5,6\}$

2-clique: $\{1,2,3,4,5\}$, $\{2,3,4,5,6\}$
2-clan: $\{2,3,4,5,6\}$
2-clubs: $\{1,2,3,4\}$, $\{1,2,3,5\}$, $\{2,3,4,5,6\}$

Figure 2.2: ([188]). Cliques, n-cliques, n-clans and n-clubs in undirected graphs.

2.1.2.2 Directed Graphs

Undirected edges can model a number of social relations such as "communicates with each other" or "are relatives", but usually social relations are directed. If A likes B then it is not automatically implied that B likes A.
If we regard directed edges in social relation graphs, we need to broaden our definitions. In case of n-cliques in a graph G we need to regard four cases [188]:

- A **weakly connected n-clique** is a maximal weakly n-connected sub-graph. Two nodes are weakly n-connected, if a semi-path of length n or less exists between them. A semi-path is a path in the undirected graph G' which results from G by replacing all directed edges with undirected edges.
- A **unilaterally connected n-clique** is a maximal unilaterally n-connected sub-graph. Two nodes v_{j_1}, v_{j_2} are unilaterally n-connected if a (directed) path of length of at most n exists from v_{j_1} to v_{j_2} *or* from v_{j_2} to v_{j_1}
- A **strongly connected n-clique** is a maximal strongly n-connected sub-graph. Two nodes v_{j_1}, v_{j_2} are strongly n-connected if a (directed) path of length of at most n exists from v_{j_1} to v_{j_2} *and* from v_{j_2} to v_{j_1}
- A **recursively connected n-clique** is a strongly connected n-clique, where for each pair of nodes the two connecting paths use the same nodes.

Essentially, a weakly connected n-clique is a n-clique which ignores the relation's directions. Unilaterally and strongly connected n-cliques are plausible analogous definitions of n-cliques in the directed case and recursively connected n-cliques essentially demand reciprocal (symmetric) ties in large parts of the sub-graph which again makes distinguishing the relation directions superfluous.
In chapter 4 we will apply a technique that allows to handle directed relations in an elegant way: By introducing for every type of directed relation between nodes a heuristic which maps the two directed relations to an (undirected) similarity relation we can model groups on the basis of an undirected graph. The heuristic takes e.g. the mutuality of directed communication relations into account in order to define the similarity between the two nodes (persons). Proceeding in this way demands that we regard valued relations. That is we have to use graphs with weighted edges. Weighted relations are the next more accurate formal description of social relations which usually have a strength that characterizes them (as in the case of the "likes" relation). Thus group models on the basis of weighted edges will be discussed now.

2.1.2.3 Graphs with Valued Relations

For the sake of simplicity, we will discuss only undirected valued relations. In [188], a simple proposal for investigating cohesive subgroup structures in graphs with weighted relations is discussed:
Linearly ordering the (at most $|\mathcal{E}|$) different weights $w(e_{j_a j_b})$ of the edges $e_{j_a j_b}$ gives a hierarchy of weights

$$w_0 = w(e_{j_{a_0} j_{b_0}}) \leq w_1 = w(e_{j_{a_1} j_{b_1}}) \leq \ldots \leq w_{|\mathcal{E}|-1} = w(e_{j_{a_{|\mathcal{E}|-1}} j_{b_{|\mathcal{E}|-1}}}) \quad .$$

By introducing $|\mathcal{E}|$ suitably chosen thresholds

$$c_0 = w_0 \leq c_1 = w_1 \leq \ldots \leq c_{|\mathcal{E}|-1} = w_{|\mathcal{E}|-1}$$

we can derive from the weighted graph $G = (\mathcal{V},\ \mathcal{E} \subseteq \mathcal{V} \times \mathcal{V},\ w : \mathcal{E} \to \mathbb{R})$ a set of threshold graphs $\{G_{c_0}, G_{c_1}, \ldots, G_{c_{|\mathcal{E}|-1}}\}$, which can all be separately examined for group structures based on the models described in the previous section. Each threshold graph is defined by

$$G_{c_i} = (\mathcal{V}_{c_i},\ \mathcal{E}_{c_i}) \text{ where } \forall e \in \mathcal{E}_{c_i} : w(e) \leq c_i \tag{2.5}$$

Single-link and complete link SAHN

- Input:
  Undirected weighted Graph: $G = (\mathcal{V},\ \mathcal{E} \subseteq \mathcal{V} \times \mathcal{V},\ w : \mathcal{E} \to \mathbb{R})$.
  Larger weight = stronger tie. No two weights are equal.
- Initialize (I):
  - Sort the weights
    $$w_0 = w(e_{j_{a_0} j_{b_0}}) \ < \ w_1 = w(e_{j_{a_1} j_{b_1}}) \ < \ \ldots \ < \ w_{|\mathcal{E}|-1} = w(e_{j_{a_{|\mathcal{E}|-1}} j_{b_{|\mathcal{E}|-1}}}) \quad .$$
  - Introduce $|\mathcal{E}|$ suitably chosen thresholds
    $$c_0 = w_0 + \leq c_1 = w_1 \leq \ldots \leq c_{|\mathcal{E}|-1} = w_{|\mathcal{E}|-1}$$
  - Compute $|\mathcal{E}|$ threshold graphs $\{G_{c_0}, G_{c_1}, \ldots, G_{c_{|\mathcal{E}|-1}}\}$ by setting
    $$G_{c_0} = (\mathcal{V}_{c_0},\ \mathcal{E}_{c_0}) \text{ where } \forall e \in \mathcal{E}_{c_0} : w(e) \leq c_0$$
- Initialize (II):
  Start with the initial clustering, corresponding to G_∞ (which contains no edges) where the clustering consists of $|\mathcal{V}|$ clusters each containing exactly one vertex.
- Process:
  Choose either single-link version OR complete-link version:
  - For (step $k := |\mathcal{E}| - 1;\ k \geq 0;\ k$--) //iterate over threshold graphs
    Single-Link Step (adapted from [77]):
    If the number of components (a component is a maximally connected sub-graph) in G_{c_k} is less than the number of clusters of step $k-1$, take as clustering of the current step the clustering of step $k-1$ redefined by naming each component of G_{c_k} as a cluster.

  OR
  - For (step $k := |\mathcal{E}| - 1;\ k \geq 0;\ k$--) //iterate over threshold graphs
    Complete-Link Step (adapted from [77])
    If the union of two of the clusters in the clustering of step $k-1$ forms a clique in G_{c_k}, take as clustering of the current step the clustering of step $k-1$ redefined by merging these two clusters into one cluster.

  Endfor

Figure 2.3: Algorithm for single-link and complete-link sequential agglomerative hierarchical non-overlapping clustering. (Adapted from [77])

Combinations of the group models from all the threshold graphs or, more precisely, the class of algorithms by which these combinations can be won, are usually referred to in data-mining literature as **sequential agglomerative hierarchical non-overlapping clustering (SAHN)** [77]. We will now give a short overview about simple SAHN variants and show how they correlate with the models and notions from the previous section.

Clustering algorithms aim at discovering cohesive clusters in (large) sets of data. In contrast to supervised classification algorithms, which learn class-models from given training data, clustering algorithms are unsupervised classifiers which have to discover classes (clusters) in the data sets on the basis of heuristics. Usually, a general heuristic is to formulate the clustering problem as a coupled optimization problem: Maximize intra-cluster similarity (or more general cohesion) while minimizing inter-cluster similarity (cohesion). These topics will be extensively treated in chapter 5. As a motivation for the arguments in the rest of this section we will therefore discuss only two special variants of clustering algorithms, namely complete-link SAHN and single-link SAHN.

The SAHN algorithm proceeds **sequentially** by constructing the threshold graphs one after the other and computing the clusters from the threshold graphs. It is **agglomerative**, because when computing clusters from the threshold graph G_{c_i} it does this by merging clusters computed from threshold graphs with smaller thresholds $c_j < c_i$. It is **hierarchical** because it starts with a clustering of $|\mathcal{V}|$ clusters where each cluster has exactly one element and in each step merges clusters into larger ones. Thus in each step of the algorithm, the clusters of this step are either identical to the clusters of the previous step or are supersets of the clusters of the previous step. The algorithm is **non-overlapping** because in each step, each object (person) is member of exactly one cluster.

Starting (as above) from a symmetric (undirected), weighted graph, the algorithms for single link and complete link SAHN can be formulated as denoted in figure 2.3. Note that not all steps of the algorithm necessarily change the clustering.

In order to allow for a better understanding of the algorithm we present an example adapted from [77]. Consider the weights of the edges in a graph of five nodes given by the following weight-adjacence-matrix A:

$$A = \begin{array}{c} \begin{array}{ccccc} v_0 & v_1 & v_2 & v_3 & v_4 \end{array} \\ \left(\begin{array}{ccccc} \infty & 4 & 2 & 8 & 3 \\ 4 & \infty & 9 & 5 & 7 \\ 2 & 9 & \infty & 0 & 1 \\ 8 & 5 & 0 & \infty & 6 \\ 3 & 7 & 1 & 6 & \infty \end{array} \right) \end{array} \begin{array}{c} v_0 \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} \tag{2.6}$$

This matrix leads to the threshold graphs shown in figure 2.4 (not all threshold graphs are depicted because we only need them down to graph G_3).

Following the algorithm in figure 2.3 leads to the following hierarchical clusterings which are depicted with the help of dendrograms in figure 2.5.

The presented algorithms use variants of two group models from [188]. In case of the **complete link** algorithm a variant of **cliques** is used, where two connected nodes also are allowed to form a clique (in [188], a clique consisted of three nodes at minimum). In contrast to the proposal in [188], not all cliques of the threshold graphs are considered as groups (cliques) in the complete link case. For example the clique $\{v_1, v_4, v_3\}$ in G_5 is not considered as a cluster. Once the complete link clusters $\{v_1, v_2\}$ and $\{v_0, v_3\}$ have been established, they cannot be broken up

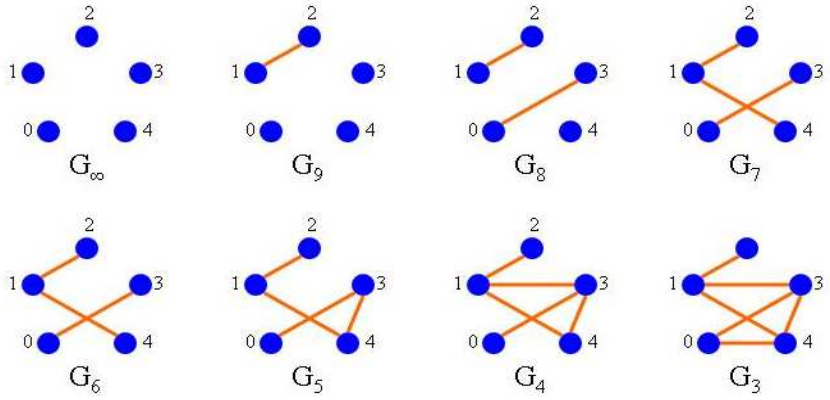


Figure 2.4: (Adapted from [77]) Threshold graphs from the graph characterized by weight-matrix A in (2.6)

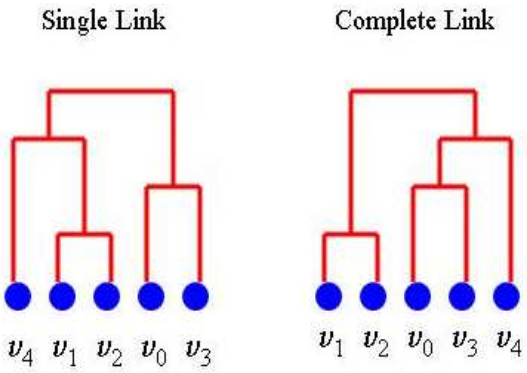


Figure 2.5: (Adapted from [77]) Dendrograms resulting from applying the algorithm of figure 2.3 to the threshold graphs from figure 2.5

again and node v_4 must join either $\{v_1, v_2\}$ or $\{v_0, v_3\}$ [77].
In case of the **single link** algorithm, a relaxed variant of **n-clubs** is used. As has been discussed in the previous section, an n-club is a maximal sub-graph with the probability that every node is connected to any other node via a path of length $\leq n$. The single link algorithm takes maximal connected sub-graphs as candidates for groups. Thus the groups found by single link clustering are not as strictly defined as in the case of n-clubs in threshold graphs.

2.1.3 Conclusions

Based on what has been said in the previous sections, we can conclude the following aspects concerning groups and group models:

1. **Clustering-Results as Group Models.**
   Sociometry approaches based on emergence and individualistic schools of thought, model

groups as substructures in the graph of dyadic social relations in a society (set) of people, as has been explained before. Common sense and common experience suggests that social relations should be modeled as **valued relations**. (Compare [46]). The strength of a social relation can be expressed by e.g. the number of communication acts in a communication relation or a degree of sympathy in a sympathy relation. As we have shown in the previous section, certain variants of common group models from sociometry can also be derived from such weighted graphs with the help of **clustering algorithms**. It is a reasonable assumption that group models obtained by applying other forms of clustering algorithms are valid group models as well. Since this is a central conclusion for this thesis, we will elaborate on this point later in this chapter in more depth. Clustering can take discrete, weighted, explicit graph representations of social relations as input which correspond more to a individualistic point of view. But they can also operate on property data of the group members which have a well defined distance measure which reflects more a collectivistic point of view. This assumption / conclusion is backed by social identity theory and many other studies which stress that similar properties and characteristics of people facilitate group formation (see discussion in 2.1.1.2 and before) and that groups have a tendency to unify general properties of their members. Thus it is reasonable (although not automatically guaranteed) that **similar properties are indicators for group-membership**. When applying this conclusion, further heuristics need to be taken into account in order to avoid investigating **quasi-groups** (see 2.1.1) that have no real social significance. This point will also be extensively discussed throughout the rest of the thesis.

2. **Regularity and Periodicity**.
   Social psychology indicates that a group is characterized by **face-to-face-interactions** which have a certain **duration**. It can be concluded from common sense and common experience, that socially relevant **group-interactions** are often characterized by **regularity and periodicity**. That means that group interactions very often occur repeatedly and that these repetitions very often follow a periodic pattern. Cultural, biological and physical boundary conditions of our existence imply that almost all aspects of life are periodic. E.g. in international culture, time-management is strongly influenced by organizing time according to periodic patterns like weeks or months. This has profound effects on working time schedules as well as on spare time schedules. E.g. the practice times of a basketball team need to follow a periodic pattern in order to allow for a coordination of the team members. Since the majority of all professional and spare time activities are bound to periodic patterns, even those group interactions that would on the first glance not need to be bound to such a scheme (such as times spent with the family) are indirectly influenced by the periodic schemes as well.

3. **Spatio-Temporal Proximity**.
   As we have seen, many studies show that space has a profound influence on groups. Social Impact theory suggests that the social impact of persons on one another decreases quadratically with the distance. Since the importance of face to-face interactions is strongly emphasized in all characterizations of groups (see discussion in 2.1.1) we can conclude that **spatio-temporal proximity** is a good **indicator for a group**. This argumentation a special version of the conclusion (1) in this section. Naturally not all spatio-temporal clusters of persons are socially relevant groups. As an example, consider a number of people standing in line for something. We will see in later chapters how this criterion can be complemented with other heuristics in order to detect and model groups.

4. **Cut-Off-Distances.**
   As stated in section 2.1.1.1, **space imposes constraints on interaction patterns in groups**. (compare the citation which states that 30 meters may represent an order of magnitude for an interaction "cut-off" distance). As we have seen, many researchers in social psychology emphasize that face-to-face communication is the key factor in group interaction. Although the notion of "face-to-face" is not precisely defined in literature and although no studies could be found that systematically investigate the spatial limits for immediate group interaction, common sense and common experience show that **cut-off distance limits** exist. "Hardware" limits of our sense organs are one factor: Our eyes allow for an identification of other persons in a distance of roughly 50 to 100 meters and in an angle area of at most 30 degrees [196]. In urban environments the maximal distance of interaction is much more severely limited by e.g. limited lines of sight (e.g. because of buildings) and e.g. by the presence of many other people. Since our psycho-perceptional filter systems are very well adapted to context, only a very close environment is usually perceived socially and the rest is filtered out by the brain. The same is true for the aural sense for normal conversation which in crowded settings reaches from only a few meters to at most 10 or 20 meters.
   Apart from the sense limitations, cultural and psychological boundary conditions also determine distances in group interactions. E.g. in each culture, certain conventions exist concerning which interaction distances are perceived as polite.
   Since these limitations usually equally apply to all members of the group, we would expect that **spherical or elliptic spatial distributions** of group members (clusters) are more common than linear or very elongated distributions.

2.2 Groups in Communication- and Information-Management

Social identity theory suggests that a substantial part of our social identity is formed by our involvement in groups. In terms of communication- and information-management groups play an important role in information technology too.
Support for **teams** as special forms of professionally oriented groups with a clear working goal is extensively investigated in **CSCW**. Distributed working environments provide special distributed editors and workspaces for collectively and collaboratively manipulating documents, source code or other working artifacts. As an example, consider the BSCW shared workspace system [141] or even the CVS distributed versioning tool [142]. In these systems, access control is usually implemented on a group-level granularity.
Virtual communities are regarded by some researchers as groups too. Extensive research on supporting communities has been conducted (see chapter 1). Since we will usually regard groups as rather limited in extension, we will take a more detailed view in the next section by investigating group structures **within** communities.
If we perceive **computer games** as a means of communication, we can find collaboration and groups in a wide variety in the online gamer's world. Examples are groups in MUDs (Multi-User-Dungeons) or Quake-clans (groups playing the 3D-action game Quake).
Information published on the **Web** is often associated with groups in multiple ways. Groups can occur as authors of documents and trust in the published information is often directly associated with the reputation of the group that publishes it. As an example consider scientific papers or articles in online magazines authored by a group of authors.

Instant-Messaging (IM) environments like AOL Instant Messenger [143] are the most prominent example of **group-level electronic communication** on the Web. By determining a personal **buddylist** the IM environment establishes from the point of view of a single user an n:1, nearly synchronous, text-based communication channel to the buddies. The buddylist acts as a group-level filter that determines the message generators that a user is willing to accept messages from. If the buddylists from some members significantly overlap, we can speak of a primitive form of community with a volatile CIKS.
Instead of going further into details about groups in information- and communication-systems in general, we will now take a closer look at groups and group-level support in communities especially in mobile communities.

2.3 (Ad-Hoc-)Groups in Mixed-Real-World-Virtual Mobile Communities

From the discussion so far and from the conclusions drawn, we will now investigate the role of groups in **mixed-real-world-virtual mobile communities** and will formulate an agenda that states how we will detect, model and utilize groups and their manifestations in these communities. (For the sake of brevity, we will abbreviate mixed-real-world-virtual mobile communities shortly by MMC). The main notion of this section will be the notion of an **Ad-Hoc-Group**.

2.3.1 Characterization of Ad-Hoc-Groups

In the previous sections it was not clearly distinguished between social groups that exist in an "abstract" way (e.g. as perceptional representations in the minds of their members or as organizational units) and the concrete spatio-temporal manifestations of these "abstract" groups. A typical example is a family. It is a social group whose de-facto existence no family member would deny. Although a family is in most cases in a spatio-temporal situation where all members are far away from each other and no communication takes place, the family will still be perceived as an existing group. We will call this form of existence **abstract existence of a group**. If family members meet and interact in a "place" in space in time (which we will call a **spatio-temporal situation**) the group manifests itself (or as we could also say comes into function or is instantiated). We will call such an instantiation or manifestation an **Ad-Hoc-Group**. Over the course of time, an abstract group can mainfest itself multiple times as an Ad-Hoc-Group. For example, a family (abstract group) will meet every year on christmas. The concrete instantiation of the family meeting in a particular year in a particular place (concrete spatio temporal situation) is then one Ad-Hoc-Group. Each Ad-Hoc-Group instantiation of an abstract group can thus be tagged by the concrete spatio temporal situation it manifests itself in.
The notion "Ad-Hoc-Group" was chosen, because we will often have situations where single spatio-temporal instantiations are the only form of existence of a group. That means that the group will only manifest itself in one or a very small number of spatio-temporal situations and that its abstract existence is more or less confined to these situations. As an example consider a group of people meeting in a compartment in a train, talking to each other during the trip and then parting without ever seeing each other again. Another example is a group of boys from different parts of a country spending their holidays in a pathfinder camp.
In virtual communities, groups with abstract existence (e.g. a circle of hobby philosophers

that meet every Tuesday in a chat-room to discuss Existential Philosophy), may also manifest themselves in time (e.g. every Tuesday) but will manifest itself in **virtual space** in contrast to physical space. Thus we will call the situations in which such **virtual manifestations** occur, **virtual-spatio-temporal situations**.
We will define the notion of spatio-temporal situation precisely in chapter 3.

2.3.2 Ad-Hoc-Groups in MMCs

Do Real-world social **groups** and their spatio temporal manifestations (Ad-Hoc-Groups) play an important role in MMCs and can they be used as a **modeling paradigm** for information management in those communities? The following arguments are intended to support a positive answer to these two questions.

In **virtual communities**, groups are very prominent. Access to private parts of the community's CIKS is often organized on the group level. In most cases, **buddylists** are the usual way to group community members into an in-group (the buddies) and an out-group (the rest). In the COSMOS project, we have centered privacy aspects (such as accessing certain profile elements) and communication services around the concept of buddylists and shared buddylists (e.g. recipients for Virtual Post-Its or Portable Messages are usually determined through buddylists and shared buddylists).
In large **chat based communities** (such as communities within ICQ) we find that groups create their own chat-rooms and that diversification into sub-groups is a natural process. Furthermore, when investigating communities whose main communication- and information medium are **discussion boards**, we find that the flow of communication and information is also often controlled by groups: Certain sets of people support one opinion and other groups support other opinions. (Compare the phenomenon of flame-wars [31]).
In view of groups in **MMCs**, we have made the observation that due to the deep and context-sensitive embedding of MMC community services into the life of the members, the **real world social relations** between the members play a more important role than in case of purely virtual communities. As has been explained before at the beginning of section 2.1 the communicative boundaries of the communities become more fuzzy and community members may use the community services in social situations which involve group **interactions** with community (platform) members as well as **with non-community (platform) members**.
As a consequence, the **part of the social net** that involves non-community (platform) members is **not directly reflected** in the community's CIKS. In order to adequately support community members in a socially context sensitive way with information and communication services we need to focus on social structures which are easy to detect and which have a high probability of reflecting social structures which may also reach outside the community platform's borders. Groups are such social structures. Because of the high degree of identification and cohesion of group members within their group, they are the easiest structures to model and to characterize even in cases where parts of the group are not reflected in the community CIKS. In case of dyadic relations it is impossible to measure such a relation, if one partner is outside of the community. (This situation might be different in a completely distributed agent-based model, where no central community platform exists (see [46])). E.g. if we take a group-level granularity view, the chance that the content of an information filtering service event reaching a group of community members is relevant even for the group members outside the community is high if the part of the group within the community is large enough to characterize the group.

Another aspect which makes groups interesting as targets for socially context sensitive community services is the phenomenon discussed in section 2.1.2 which leads to **group-think**. As we have seen in 2.1.2, we can find a tendency in groups to **unify opinions, goals** and even **acting**. In terms of context- and social-sensitive support for groups with information- and communication services we can profit from these normative effects. Since the group strongly influences the opinions, goals and way of thinking of group members we can build our services on models of the **group as a whole**. These models can be obtained on a much broader data-basis than in case of single users. Algorithms that make use of this phenomenon are e.g. **Collaborative Filtering**.
In case of **Ad-Hoc-Groups** we can go one step further: if we make groups targets of our community services while they manifest themselves in a spatio-temporal situation (as an Ad-Hoc-Group) the chance of being not only socially relevant (as in case of conventional Collaborative filtering) but even **contextually-socially relevant** is high. As an example consider a proactive collaborative filtering algorithm that analyzes groups based on their interests. If it finds a group that is interested in e.g. soccer, it is socially relevant to forward the soccer results to every member of the group. It is even contextually-socially relevant if the soccer results are forwarded when the soccer-group **meets as an Ad-Hoc-Group**. We will further elaborate on aspects of the concept of contextual-social relevance in chapter 6.

2.4 Detecting and Modeling Ad-Hoc-Groups

If we accept groups as valuable modeling paradigms for the support of mobile communities with mixed real-worl-virtual character, how do we **detect, model and use groups in such MMCs**?

- In terms of detecting and modeling groups in MMCs we have found that, inspired from sociometry, **clustering procedures** deliver good candidates for groups (Compare conclusion (1) from section 2.1.3). From our other conclusions of section 2.1.3 we will derive what the objects from a community's CIKS should be that we apply these class of algorithms to. We will also derive from these conclusions what the heuristics are that determine the choice of parameters or other special features of the clustering algorithms used.

- Applying conclusion (2) from section 2.1.3 to the notion of **Ad-Hoc-Groups** we find that **Ad-Hoc-Group** manifestations in time and space can be a good indicator for the **structure of the underlying abstract group**.
  We can broaden this conclusion by saying that clusters found on the basis of highly dynamic parameters (profile items) of people (the **context**) may be candidates for Ad-Hoc Groups. Time and Space are the most prominent examples of such context parameters.

- Applying conclusion (2) and (3) we find that clusters which form on a **regular temporal and / or spatial basis** are better candidates for Ad-Hoc-Groups (manifestations of real abstract social groups) than clusters which form only once.

- Applying conclusion (1) and what has been said about social impact theory and social identity theory we find that clusters of **similar personal parameters** (such as interests) are candidates for abstract groups. We have to apply suitable heuristics in the clustering process to avoid finding quasi groups without social reality.

- If we perceive Ad-Hoc-Groups as contextual manifestations of abstract groups, it is reasonable to assume that we can, **vice versa**, gather properties of the Ad-Hoc-Group from properties of the abstract group which are found through clustering over less dynamic attributes (e.g. interests or long-term communication behavior))
- We assume that we can conclude from the regular and long-lasting existence of an Ad-Hoc-Group the existence and structure of its abstract group (through clustering of contextual parameters or social relations) and from properties of the abstract group conclude properties of the corresponding Ad-Hoc-Group (through clustering of parameters or social relations with low dynamics). We conclude from that that we will obtain better models for groups if we **intersect the results** from both types of clustering (contextual and non-contextual)
- From conclusion (4) we draw a **further heuristic for contextual clustering** in space and time: Certain cut-off values should be introduced to rule out clusters which cannot be candidates for Ad-Hoc-Groups because their diameter exceeds socially and psychologically induced maximal distances. The cut-off values should in the ideal case depend on the overall density of persons. (See chapter 3 for a deeper discussion).

These points induce an **agenda** for the rest of this thesis. First we will devote some space for the thorough investigation of space and velocities as examples for contextual parameters. We will develop a similarity measure based on the above points and will develop a stochastic simulation for realistic movement of people in urban areas. We will then investigate what types of less dynamic relations and parameters from a MMC's Information and Knowledge Space we can use for characterizing abstract groups. For two example types of data, we will develop sophisticated similarity measures with respect to the later clustering applications. We will then discuss the application of clustering for the detection and modeling of Ad-Hoc-Groups and abstract groups, develop suitable heuristics for applying these procedures with respect to the points from above. At last, some example application fields for the obtained group characterizations in view of information management in MMCs will be introduced.

Chapter 3

Contextual Data: Locations and Velocities

This chapter discusses data sources for the modeling of ad-hoc groups in mobile communities in general and location and velocity data in particular. The first section gives an overview on classes of data that can be used for user-modeling and on what additional or different possibilities for data sources a collaborative community information- and knowledge space can provide. We identify three major classes of data sources of special importance for the given task: Contextual data, explicit self information data and implicit textual communication data with tree-like structure. The latter two will be dealt with in the following chapter. As an example for contextual data, locations and velocities are investigated. We discuss, how location and velocity data can be collected and we discuss some essential privacy and pragmatic issues in connection with location data. Since location and velocity data needed for the investigations of later chapters are not available in the required precision at the present time, the following part is devoted to the development and extensive discussion of abstract stochastic models for the simulation of such data. The implementation of these models in the SUMI simulation toolkit is introduced. Finally, similarity measures for location and velocity data are investigated in view of the task of identifying Ad-Hoc Groups.

3.1 Data for User-Modeling

Recent years showed a growing interest in applications that have a high degree of personalization and context sensitiveness especially in the field of web-applications [44]. As has been terminologically characterized in chapter 1, we can define **personalization** as a high degree of incorporation of static user profile information and **context-sensitiveness** as incorporation of dynamic user profile information.
On the basis of user-interaction, many types of information can be gathered that can be used to build user profiles (user models). For example in an E-Commerce web-application, action-to-item-affinities, item-to-item-affinities and user-to-user-affinities can be learned from collecting a user's interactions with the application [44] such as browsing sequences, queries, purchase actions etc.. As an example, action-to-item-affinities can be learned by monitoring a user's queries with the site's internal search engine and matching them against the spectrum of offered items. Item-to-item-affinities can e.g. be concluded by monitoring a user's purchase history and matching the bought items against the spectrum of offered items. User-to-user-affinities can e.g. be inferred by comparing browsing histories on product sites between users. The deduced

affinities can be stored as components of a user's profile and can be used for personalization by e.g. suggesting further items on a user-to-user-affinity or user-to-item-affinity basis as in case of Amazon bookstore [133].
In this section we will briefly review possible general data sources for existing personalization techniques, examine what additional data can be gathered or evaluated in a collaborative environment like a community and what further possibilities a mobile environment provides especially for the analysis of Ad-hoc-groups.

3.1.1 Data for Individual User-Modeling

When discussing data types and sources that information for the build up of user profiles can be gathered from, the first aspect to consider is the application that the user profiles will be used for. A classic field of application of user profiles is information filtering. Information filtering is essentially a classification problem where on the basis of past individual or collaborative interaction with an information system a model is learned that allows to decide for a individual user or a group of users and a given information item whether this item is interesting or not (or the degree of interestingness of this item respectively). Recommender systems on the basis of collaborative filtering are well known examples (see e.g. [156]). Without antedating the discussion in chapter 6, we will assume information filtering as the main application for the present discussion.
The previous example of affinities in E-Commerce shows that it is necessary to thoroughly distinguish several notions and concepts when talking about data for user-modeling. What can actually be collected from a user is raw data which can then be interpreted as information or from which information or knowledge can be extracted or inferred with the help of data-mining heuristics. We can distinguish between **implicitly collected** data and **explicitly collected** data. Implicit data collection does not involve any user action that is causally linked with the data collection process. The user's behavior is monitored automatically. Explicit data collection is essentially achieved by asking a user directly for the desired information.
The collected data can be on different **scales of measure** [161] such as nominal scale (only operation: test on equality (e.g. words in a text)), ordinal scale (additional operation: $<$ (e.g. ratings on a "good" "average" "bad" scale), interval scale (additional operation: $+$ (e.g. annual details such as 1999)) and proportional scale (additional operation: multiplication (e.g. time-duration in minutes)).
Examples for explicitly collected data include single valued statements such as names, ages, and telephone numbers [88], furthermore ratings on an ordinal scale [27] and choices for an option, product or category [152, 106].
Implicitly collected data for the build up of individual user profiles for information filtering and recommendation systems include mouse activities (dragging, scrolling, clicking etc.) [27], browsing and bookmarking statistics and analysis of link structures on the visited pages [108, 25], time spent reading an information item [84, 120] and counting interaction events on the file level (editing, saving, opening etc.) [148]. Textual content such as content of user communication, content of bookmarked pages etc. which can be viewed as ordered sets of nominal scaled data (words) is also a valuable data source [51] either as an explicit or implicit data source. Often the models induced from these data are compared with explicit ratings from the users to judge their quality.
After the data have been collected they have to be **preprocessed**. In case of text data such as e-mails, newsgroup postings etc. this includes e.g. stemming, stop-word removal and spelling

correction (see [51, 6]). In case of numerical data pre-processing can e.g. involve normalization, interpolation of missing values etc. (see [161]).
The collected raw data is then processed into a model with the help of a set of **heuristic** algorithms [161, 58] such as statistical algorithms (e.g. correlation or regression analysis), classification algorithms (supervised and unsupervised (clustering)), decision trees, rule-systems (e.g. inductive logical programming) etc..
The spectrum of possible data sources, preprocessing techniques and heuristic models for information filtering is so vast that it is not possible to give an overview here. The reader is referred to standard literature on data mining and artificial intelligence and chapter 5 where some of these techniques will be investigated in more detail. Instead we will now more thoroughly examine what possibilities and data a collaborative information- and knowledge space of a community provides for user modeling or profile generation respectively.

3.1.2 Data for User-Modeling in Communities

In chapter 1 we have narrowed our domain of discourse to virtual communities, whose common pursuit can be identified with the build up of a collaborative information- or knowledge space (CIKS). It has been motivated that this modeling view can be applied to a wide variety of communities and such spaces have been generally characterized.
In terms of user modeling, a CIKS is a very valuable data source. Since the CIKS and its associated communication services are often bound to a single platform, data collection and model generation can be achieved in a convenient way within this community platform under a single privacy model. Even if the CIKS and the services are distributed, trust and social coherence within the community allow for more sensitive types of data to be used and more accurate models to be created because the generated models will still be exclusively associated with the community and will be used for the community's common pursuit only.
Existing User profiles contain many explicitly collected elements such as lists of interests, names, ages etc. which directly represent usable user models or allow for the construction of formal models with little effort. Information items contain communication content and meta data that is especially valuable for modeling user interests and deducing dyadic social relations among community members [46]. If explicit representations of relations exist, they are also enormously valuable for user modeling and also already represent almost ready-to-use-models. User-item-relations can e.g. be used for collaborative filtering and user-user-relations can e.g. be used for expert finding, visualization of social networks and deduction of further user-user-relations [46, 81, 115].
Profiles, items and relations represent already rather complex high level models themselves. In essence, they represent only a modeling view of a community. What is more important in terms of raw data sources for modeling users in a community is that the comparatively dense net of social relations within a community and the strong orientation towards communication and the formation of a CIKS gives access to a broader spectrum of interpretable data than in case of a single user interacting with an information system. Trust and communication needs motivate people to built extensive profiles that are intended to inform other interested users about them. In e.g. an E-Commerce setting, people tend to be much more cautious about giving access to personal information than in a trusted community. Communication patterns and content are more easily accessible within a community. Furthermore, communities are places where relations such as user-item-relations are made explicit by the users to support other user's information needs. Also explicit representations of user-user-relations such as Buddylists usually show up

more frequently in communities.
Because social structures are so clearly represented in a community's CIKS, it is possible to model not only individual users and their isolated views on e.g. information items, but to model whole groups of users and their interaction.
Two typical classes of data sources that occur frequently in CIKS are explicit data, especially **explicit self information** data and implicit data, especially **tree-like forms of textual communication data**. Explicit self information data have the primary function of directly informing other users about oneself. They represent special forms of textual, asynchronous, 1:n communication and are an essential part of a person's profile and are characterized as being explicitly collected, often as answers in a profile generation form. They are interesting, because the questions that induce these explicit self information answers limit the semantic scope of these data and allow for special adapted heuristics to be applied. Usually each answer has a very limited semantic extension and often corresponds to a single concept or at least to a coherent piece of information. Furthermore, data that users provide about themselves can be considered the most valuable form of data that can be used for revealing social structures like Ad-Hoc Groups and information filtering tasks in general. As an example we will investigate lists of interests which show up in various profiles and virtual business cards in communities [131, 55]. They are interesting for the information filtering task because every element (text-phrase or single word) of such a list usually corresponds to a single concept of interest. We will investigate these vectors in detail in chapter 4.
The second source of information are collections of tree-like communication data like discussion boards or net-news. They are collected implicitly and represent typical structures of a CIKS. They are means of indirect, n:m, text-based communication which has been identified in the first chapter as one of the most typical forms of information exchange in virtual communities. Content and tree structure of such discussion boards reveal interesting details about interests and social relations among community members. We will deal with this data source in detail in chapter 4.
We will now take a closer look of what additional contextual data sources a mobile community provides.

3.1.3 Data for Modeling Ad-Hoc-Groups in Mobile Communities

As has been discussed above, a community's CIKS contains many data sources that reflect dyadic (binary) and group relations among the members of a community. In a pure virtual community the dyadic and group relations are virtual too. That means they usually do not have a correspondence to real world relations. In a mobile, mixed real world - virtual community, the virtual relations correspond to and are augmented and influenced by real world social relations. Therefore the (artificial) distinction between virtual and real social relations can be dropped in case of mobile communities. In order to analyze social relations and especially social group relations that have aspects of real world interaction, we can access contextual information (highly dynamic user profile elements). Context information reveals short lived hints for social relations that may manifest themselves only in short periods of time. Such contextual manifestations of social group relations or group relations that actually only exist for such comparatively short periods of time have been called Ad-Hoc-Groups in the previous chapter.
Types of contextual information that can be used for user-modeling and especially for the characterization of Ad-Hoc-Groups include location and velocity information and directions of view. Directions of view have been used in tourist information and augmented reality systems in the

past to detect which objects a user is visually focusing. The Cyberguide system [4], the GUIDE system [28] and the Websign system [153] have made attempts towards using viewing direction information. Under the heuristic that an object that is focused for a significantly long period of time is interesting for the focusing person, the information system can proactively provide information about that object. If several people focus one object, it can be inferred that they are collectively interested in the object and assuming the heuristic that several collective focusing episodes within a short period of time indicate a social Ad-Hoc-group relation it could be inferred that these people are members of an Ad-Hoc-group. An example are people involved in a guided tour in a museum. These directions can be measured with appropriate augmented reality hardware, which is unavailable in appropriate numbers in projects like COSMOS. Although viewing directions are an interesting data source we will not further elaborate on this subject. Locations and velocities are excellent data sources for modeling individual users and groups of users. Location based services represent the most discussed examples for context-sensitive applications as has been pointed out in chapter 1. If people share the same location and / or move in the same direction with the same speed it can be heuristically concluded, that they form an Ad-Hoc-Group. We will further elaborate on this subject in chapter 5.

In the following sections we will investigate locations and velocities as a typical example for contextual, highly dynamic information or data. Collection process, simulation possibilities, test-collections and similarity measures with the goal of identifying Ad-Hoc-groups will be discussed in detail. The actual Ad-Hoc-group detection algorithms will then be discussed in chapter 5.

3.2 Accessing Localization Information

As has been pointed out in section 1.4.2, location information can be won by a great diversity of location technologies. The accuracy of these basic methods can be improved by several measures. E.g. one possible augmentation for the cell based approach is triangulation with the help of neighbor cells. The left part of figure 3.1 depicts the basic principle: The mobile device estimates the distances to two neighbor cell base stations and the distance to the base stations of the cell that the device is booked in with the help of field-strength measurements. If the locations of the base stations are known, simple geometry calculations yield the location of the mobile node. Unfortunately, the estimation of the distances is very coarse grained because it is used in the GSM protocol for signal time-multiplex-slot timing measures only which does not require great accuracy. Therefore, triangulation can only increase the accuracy by a factor of 2 [3, 7]. Furthermore, as has been pointed out in 1.4.2 not all technologies work in all spatial situations: GPS does not work indoors, access points and beacons have a limited range etc.. Availability and accuracy limitations introduce a trade off hierarchy of location technologies which can be roughly compared to the memory hierarchy in a computer system: since there is no memory technology which is cheap, fast and persistent at the same time, a trade off between these factors has to be found. At the present time no single, publicly available location technology is able to deliver location information with an accuracy and spatial availability which would allow for a continuous mobility profile of a single person or a group of persons to be captured. The cell based location technology used in the COSMOS project which is depicted in the right part of figure 3.1 suffers from further limitations, because there is no official support for the access of cell based localization information within the operating system of the mobile device which makes it necessary to use the complicated scheme shown in the figure which is very error-prone due to delicate configuration, unpredictable and frequent hand-overs between base stations without

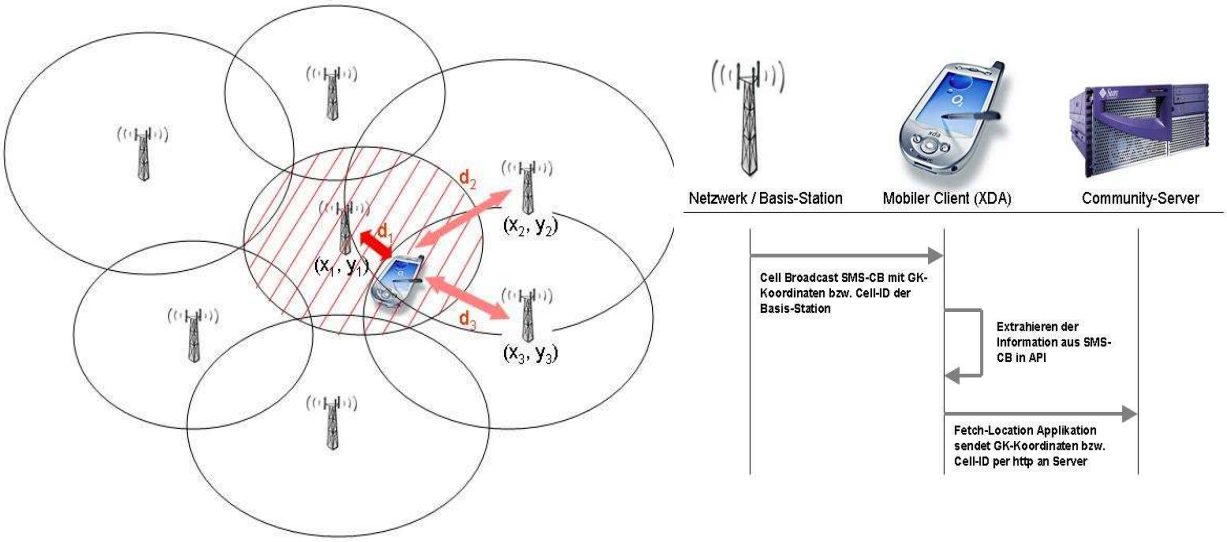


Figure 3.1: **Left figure:** Advanced Positioning through additional triangulation calculations. Requires knowledge of the positions $(x_1, y_1), (x_2, y_2), (x_3, y_3), \ldots$ of the base stations and the distances $d_1, d_2, d_3, \ldots$ to the base-stations which are coarsely estimated through field-strength measurements [181]. **Right figure:** Event-based, proactive transfer of the localization information to the community server which keeps track of the recent locations of all users [181].

moving the mobile device and possible loss of GPRS connection.

With cell based localization of an accuracy of roughly $(200m)^2$, velocity estimations would be extremely coarse grained even without these further limiting factors, because it could only be estimated by the time between cell-handovers and their spatial distance. Since these handovers are only very coarsely related to spatial movement, velocity estimations are virtually impossible.

It is fair to assume that some or all of these limitations will be overcome, when a cooperative hierarchy of location techniques will be available in the near future. For now, the only chance to acquire location and velocity data with suitable precision over a continuous time interval for several persons in parallel is a stochastic simulation which will be a major focus of the remaining part of the chapter.

3.3 Privacy and Pragmatics of Location Data

Although privacy issues are not in the focus of this thesis, some words about privacy and pragmatics of location data are necessary in this context.
Clearly, all types of personal information in a CIKS, especially those which were subsumed under profile information in the modeling view of chapter 1 need protection against unauthorized access. This includes indirect access via such services as the location based community services described in chapter 1.
In conventional individual location based services such as location based information retrieval (e.g "where is the next filling station") or location based telecommunication billing (e.g. O_2 Genion in Germany), the location information is either transferred only implicitly (as in case of the Genion billing) or is uniquely transferred together with e.g. a query and not permanently kept at a central server. In each case locations are in general not accessible for others at all.

In case of location based community services such as a friend-finder service (see chapter 1) the situation is much more complicated because usually we have a central service providing entity (a community server) where the locations of all users have to be collected and stored at all times. Furthermore, location data are explicitly and implicitly accessible by others.
According to the principle that a user must have complete control over his personal information at all times, such an "Orwellian" scenario of having his personal location monitored at all times is difficult to match with an unobtrusive mobile user interface, where a user is not likely to tolerate pop-ups at any time a location is updated asking for permission. The only mechanisms that make such a scenario possible are trust in the community and rule guided policies that govern the direct and indirect access to location information. Such policies allow for a precise control over location information while at the same time do not require obtrusive permissions every time location information is transferred from the mobile device. The COSMOS platforms allow for rules limiting the access to personal location information to different precision levels and sets of persons. See [88, 50] and chapter 1 for suggestions on how these policies can be implemented. An interesting approach for privacy control in decentralized architecture where each profile is maintained by a single user agent is discussed in [194].

As has been mentioned before, location and velocity information of a precision required for Ad-Hoc-Group Analysis and most other location based community applications is not available on a larger scale with the present mobile devices and infrastructure. Nevertheless, in order to investigate the algorithms proposed in chapter 5 we need sample data, which can only be obtained through stochastic simulation. We will therefore in the next sections investigate stochastic models for location and velocity data.

3.4 Existing Continuous Mobility Models

Mobility models for stochastic mobility simulations can be found in the fields of traffic management research and network research. Traffic management research simulates vehicles moving along predefined paths (roads). Well known simulator packages are GloMoSim [199] and NS-2 [132]. Since we do not primarily deal with vehicles as mobile agents and individuals are not strictly bound to moving along constrained paths, we will not go into detail about these simulators (see [92, 93] for interesting applications of GloMoSim in the field of ad-hoc-networks for vehicle-to-vehicle communication combining aspects of traffic management research and network research).
In order to motivate the choice for the mobility models used here, we will rather critically review some fundamental continuous mobility models for individuals and groups, where the nodes (individuals and groups) are not bound to move along predefined paths. Continuous models are far more easy to handle and it is reasonable to assume that they represent a good approximation of individual motion in dense urban areas.

3.4.1 Individual Mobility Models

For the unconstrained simulation of individual mobile nodes mostly used in mobile network protocol research, several simple stochastic models exist (see [22] for an overview).

Random Walk is a very simple model which is also sometimes referred to as "Brownian Motion" and which goes back to Einstein (1926) (see [22]). At fixed time intervals or after a

fixed traveled distance, a new velocity is randomly chosen in the interval $[v_{\max}, v_{\min}]$ and a new direction is chosen in the range $[0, 2\pi]$ (the basic model uses a uniform (rectangle) distribution $\mathcal{R}(a, b)$ with Riemann density $f(x) = 1/(b-a)$ if $a \leq x \leq y$ and $f(x) = 0$ else). If the boundaries of the simulation area are reached the node is reflected back.
Since velocity and direction values at step t do not depend on the values of step $t-1$, a qualitatively very erratic movement with lots of sharp turns occurs which is suitable for the simulation of e.g biological phenomena (like moving amoeba) or phenomena from statistical mechanics but is not suitable for the simulation of the movement-patterns of people in an urban area.
A Random Walk like model can also be implemented with a three node [1] Markov chain for the movement in x and y direction. The nodes of the Markov chain for x correspond to increasing the x-value by one, decreasing x by one or leaving x unaltered. Transition-probabilities are parameters of this version of the model. Although the process depends on the values of the previous iteration and thus avoids sharp turns, it does not produce continuous movements but rather movements on a grid of unit mesh size.

Random Waypoint This model is a slight modification of Random Walk. A node pauses for a certain period of time before it chooses a destination in the simulation area. The node travels to the destination at a randomly selected velocity in a straight line before pausing there again. Although the model seems to be straightforward, Camp et al. [22] point out some systematic statistical problems occurring during the start phase of a random waypoint simulation when points are initialized with a uniform random distribution. Although these problems can be overcome by cutting off the first part of the simulation and although boundary reflections can be avoided and pause times exist, the drawbacks of the sharp turns make this model inappropriate for our purpose.

Random Direction Model Here the nodes travel in straight lines until they reach the boundary of the simulation area where they choose a new direction (angle). This model was created in order to avoid density waves in the random waypoint model where nodes have a tendency to concentrate in the center of the simulation area and disperse again periodically. This phenomenon is due to the high probability of choosing a new random waypoint destination which requires a trajectory leading through the central region. Although Random Direction avoids these density waves, it's restriction on linear trajectories is inappropriate for simulating individual real world movements.

Delta-Model with Boundless Simulation Area In this model, values from time step $t-1$ are incorporated in the computations for time step t. Velocity and direction are calculated as $v(t+1) = v(t) + \Delta v$ and $\theta(t+1) = \theta(t) + \Delta\theta$ where Δv and $\Delta\theta$ are randomly chosen via an $\mathcal{R}(a, b)$ distribution. Boundary contacts are avoided by folding the simulation area to a torus by continuously continuing movements that e.g. cross a lower boundary at the corresponding point of the upper boundary. While the movement itself seems qualitatively much more suitable than it is the case with the other models introduced, the toroidal simulation area only makes sense when the individual identifier of a node does not play a role. Since this is not the case when simulating the movements of people, a torus-shaped area is inappropriate in our case.

[1] Notational remark: In this chapter, "node" can refer to the node of the graph visualizing a Markov chain and to a mobile node (individual person or group) whose movements simulated.

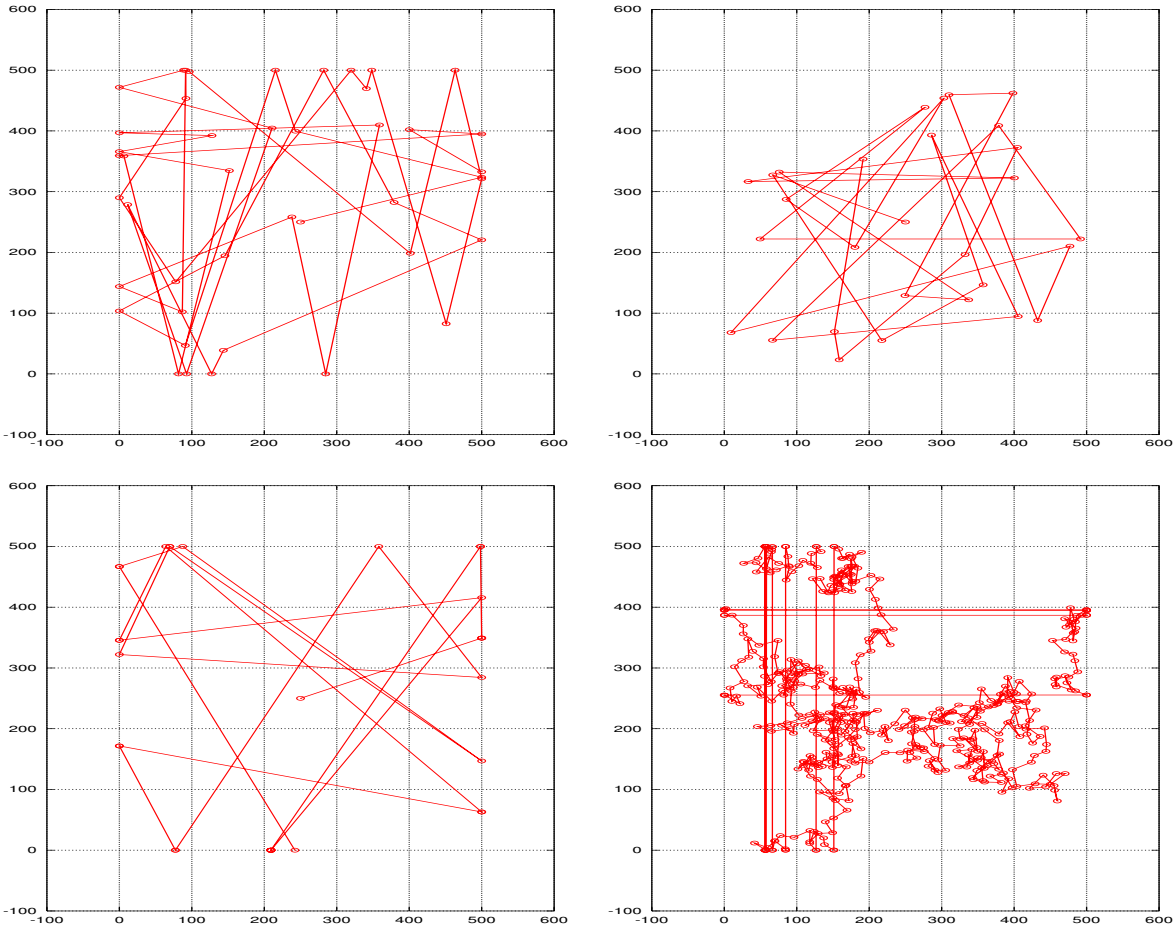


Figure 3.2: Basic Continuous Mobility Models. **Upper left corner**: Random Walk mobility model (1 node, 500 × 500 simulation-area, 1000 iterations, $\mathcal{R}_v(v_{\min}, v_{\max}) : v_{\min} = 5; v_{\max} = 15$, new calculation of velocity and direction every 50 time-steps). **Upper right corner**: Random Waypoint mobility model (1 node, 500 × 500 simulation-area, 1000 iterations, $\mathcal{R}_v(v_{\min}, v_{\max}) : v_{\min} = 5; v_{\max} = 15$, pause time: 10 time steps). **Lower left corner**: Random Direction Model (1 node, 500 × 500 simulation-area, 1000 iterations, $\mathcal{R}_v(v_{\min}, v_{\max}) : v_{\min} = 5; v_{\max} = 15$, pause time: 10 time steps). **Lower right corner**: Delta-Model with Boundless Simulation Area (1 node, 500 × 500 simulation-area, 1000 iterations, $\mathcal{R}_v(v_{\min}, v_{\max}) : v_{\min} = 5; v_{\max} = 15$ no pause times, new calculation of velocity and direction every 2 time-steps). The vertical and horizontal lines connect the points, where the node has crossed a boundary and where it returned on the opposite boundary (torus shaped simulation area). The figures where computed with the help of a collection of mobility model implementations from Camp (see [22]).

3.4.2 Group Mobility Models

In contrast to individual motion mobility models, group mobility models simulate the movements of groups of nodes for testing special aspects of network protocols. For the investigation of ad-hoc groups it is also essential to simulate sets of nodes that move together. Therefore, some basic group mobility models are reviewed. (see [74, 22])

Column Mobility Model The Column Mobility Model uses a set of reference points for a group (each node of the group has its own reference point). These reference points form a regular, rigid grid. In case of the column mobility model, the reference points are aligned along

a straight line (1 dim. grid) with fixed distances. This "column" of reference points moves as a whole: In each iteration step the new reference points are calculated from the old reference points by adding a displacement- or advance vector which is randomly chosen ($\mathcal{R}_d(d_{\min}, d_{\max})$ for its norm d and $\mathcal{R}_\theta(0, \pi)$ for its angle. So the movement of the group is basically a random walk with forward directions only. The column of reference points is either parallel to the advance vector or orthogonal to it. Individual nodes can move around their reference points via any individual mobility model described above. While this model may be adequate for groups which hold a certain distance to each other and which must adhere to a geometrical pattern when moving in a group (soldiers etc.) it is not ideal for simulating people in an urban area, where no such restrictions exist.

Pursue Mobility Model This model uses a single reference point for the group that the other nodes must follow. Positions of nodes are calculated as $x(t+1) = x(t) + a(x_{group}(t) - x(t) + r$ where a is an acceleration function, r is a random vector that adds slight (!) modifications, and $x_{group}(t)$ is the location of the group (the pursued "prey") which again can be calculated via any of the individual models from before.

Reference Point Group Mobility Model The previous two models share the simple idea of using individual models for both: computing the movements of reference points and computing displacements for the individual nodes from these reference points. This idea in its most general form is formalized in the Reference Point Group Mobility Model (RPGM) [74]. We will omit the original formalism here, because it does give insights for our purposes and because a variant of RPGM will be described in greater depth in the following.

3.5 Mobility Models Used for the SUMI Simulator

As has been discussed before, none of the previously described individual mobility models are suitable for the purpose of simulating the movement of individuals in an urban area over several days. Furthermore, for the simulation of ad-hoc groups it is necessary that individual movement and group movement occur together in the simulation and smoothly blend into each other. Groups form and dissolve again. Furthermore, people have periodic group meetings (sports-club etc.) and random group meetings, random resting times and periodic resting times (sleeping etc.) and can choose several means of transportation, each with its unique velocity distribution. It is clear that for all these aspects a realistic simulation will have to combine several suitable individual and group mobility models. For this purpose, the SUMI (Simple Urban Mobility simulator) Application was created. The application was written in Java because of easy availability of visualization tools and GUIs. The code was designed to be easily ported to C in case computing resources should be insufficient for a fast Java computation. This was accomplished by using simple data-structures (mostly Arrays), procedural coding style abandoning complicated object-oriented Structures and easy separability of Java-structures for visualization. However, on modern architectures the speed of the Java version is absolutely adequate. In the following, mobility models for each of the aforementioned aspects that are used in SUMI will be discussed.

3.6 Gauss Markov Model

In [107] a discrete version of a Gauss-Markov stochastic process was suggested for the simulation of the velocity vector v of a node in a mobile network. This mobility model was discussed and evaluated in [22] and showed an excellent qualitative performance in terms of the realism of the generated movement patterns. We will therefore shortly discuss the background of this model to provide a sound basis for discussion of modifications applied in SUMI.

Background A stochastic process is defined as a parametric family of random variables [195]

$$\{X^{(t)} \mid t \in T\} \tag{3.1}$$

where t is generally interpreted as time. T can be assumed to be an interval. A special type of stochastic processes are Gaussian processes which describe a wide variety of physical phenomena [195]. A stochastic process $\{X^{(t)} | t \in T\}$ is formally called Gaussian [195] if any linear combination

$$Z = \sum_{i=1}^{n} \alpha_i * X^{(t_i)} \tag{3.2}$$

is a Gaussian random variable ($Z \sim \mathcal{N}(\mu, \sigma)$) which implies that $\forall t : X^{(t)} \sim \mathcal{N}(\mu', \sigma')$ too.
Another class of random processes are Markov processes. A Markov process is defined in the following way [195, 47]: For any sequence $t_1 < t_2 < \ldots < t_n \in T$: we have

$$P(X^{(t_n)} < x_n \mid X^{(t_1)} = x_1, X^{(t_2)} = x_2, \ldots, X_{(t_{n-1})} = x_{n-1}) = P(X^{(t_n)} < x_n \mid X^{(t_{n-1})} = x_{n-1}). \tag{3.3}$$

That is: the Markov process is memoryless: It's behavior in the future (t_n) which is represented by the conditional probability distribution on the right hand side of equation 3.3 does only depend on the present (t_{n-1}) but not on the past ($t_{i<n-1}$). This property also seems very well suited for the simulation of velocities.
If a process is a Gaussian process and a Markov process, the process is called Gauss-Markov process. It can be shown [47] that a Gauss-Markov process that is stationary (it's statistical properties do not depend on time) is characterized by the linear differential equation

$$\frac{dX^{(t)}}{dt} + \beta X^{(t)} = W^{(t)} \tag{3.4}$$

where $W^{(t)}$ is the so called white noise random process and β can be interpreted as the degree of memory in the resulting mobility pattern [107] A Gauss-Markov process is completely described by its auto-correlation [47]

$$\phi_{XX}(\tau) = E[X^{(t)} X^{(t+\tau)}] = \sigma^2 \mathrm{e}^{-\beta|\tau|} \tag{3.5}$$

Discretizing this Gauss-Markov process $X^{(t_n)} = X^{(n\Delta t)} = X^{(n)}$ using unit time steps $\Delta t = 1$ and setting $\alpha = \mathrm{e}^{-\beta}$ one will obtain [47, 107]

$$X^{(n)} = \alpha X^{(n-1)} + (1-\alpha)\mu_X + (1-\alpha)^{\frac{1}{2}} G^{(n-1)} \tag{3.6}$$

In this equation, μ_X is the asymptotic mean of X_n for $\lim n \to \infty$ and $G^{(n)}$ is an independent ($P(G^{(n)} < g_n | G^{(n-1)} = g_{n-1}) = P(G^{(n)} < g_n)$), uncorrelated ($\phi_{GG}(\tau) = 0$) and stationary Gaussian process: $G^{(n)} \sim \mathcal{N}(0, \sigma)$ where σ is the asymptotic standard deviation of $X^{(n)}$ for $\lim n \to \infty$.
The parameter $\alpha \in [0, 1]$ can be interpreted as a measure of randomness for the discrete random variable $X^{(n)}$: If $\alpha = 1$ we have $X^{(n)} = X^{(n-1)}$ which means that X does behave completely deterministic. In case of simulating velocity and direction with such a discrete Gauss-Markov process we will obtain linear trajectories only. In case $\alpha = 0$ we get $X^{(n)} = \mu_X + G^{(n-1)}$ which represents complete randomness.

Resulting Simulation Using a discrete Gauss-Markov process for simulating the velocity can be achieved by conveniently switching to a polar representation of the two dimensional vector $v = (v^{(1)}, v^{(2)})$: v is represented by its norm $||v||$ and direction (angle) θ. For reasons of notational simplicity we will denote the norm $||v||$ simply with v and point out explicitly when the two dimensional vector $v = (v_1, v_2)$ is meant. That means that we simulate a mobile node's velocity by [22, 107]

$$v^{(n)} = \alpha\, v^{(n-1)} + (1-\alpha)\mu_v + (1-\alpha)^{\frac{1}{2}} G^{(n-1)} \tag{3.7}$$

$$\theta^{(n)} = \alpha\, \theta^{(n-1)} + (1-\alpha)\mu_\theta + (1-\alpha)^{\frac{1}{2}} G^{(n-1)} \tag{3.8}$$

The new values (v_1, v_2) for the two dimensional location vector are then

$$v_1^{(n)} = v^{(n-1)} \cos\theta^{(n-1)} \tag{3.9}$$

$$v_2^{(n)} = v^{(n-1)} \sin\theta^{(n-1)} \tag{3.10}$$

from which we easily obtain the new values for the two dimensional location vector $x = (x_1, x_2)$ from ($\Delta t = 1$):

$$x_1^{(n)} = x_1^{(n-1)} + v^{(n-1)} \cos\theta^{(n-1)} \tag{3.11}$$

$$x_2^{(n)} = x_2^{(n-1)} + v^{(n-1)} \sin\theta^{(n-1)} \tag{3.12}$$

In the simplest case, μ_v is chosen uniformly in the interval $[0, v_{\max}]$: $\mu_v \sim \mathcal{R}_{0,v_{\max}}$ and μ_θ is chosen uniformly in the interval $[0, 2\pi]$: $\mu_\theta \sim \mathcal{R}_{0,2\pi}$. Figure 3.3 shows two examples of resulting trajectories. The incorporation of previous values into the computation of present values leads to an avoidance of sharp turns and to an overall qualitatively satisfying behavior. The model was recommended in [107, 22] as qualitatively superior to the basic models of section 3.4.1. Furthermore the formalism allows for the incorporation of very useful improvements which will be discussed now.

Improvement 1: Direction-Modification The basic model does not contain any means for ensuring that the node cannot leave the simulation area like some of the previously introduced simple models have. In order to overcome this drawback, [22] suggests the introduction of a boundary into the rectangular simulation area. The crossing of this boundary by a trajectory results in a change in the value of μ_θ which softly forces the node back into the simulation area. While no analytic justification for this measure exists, the experimental results show the desired effect while qualitatively not exhibiting any unwanted features in the trajectories. From the fact that the Gauss-Markov process is memoryless, these changes

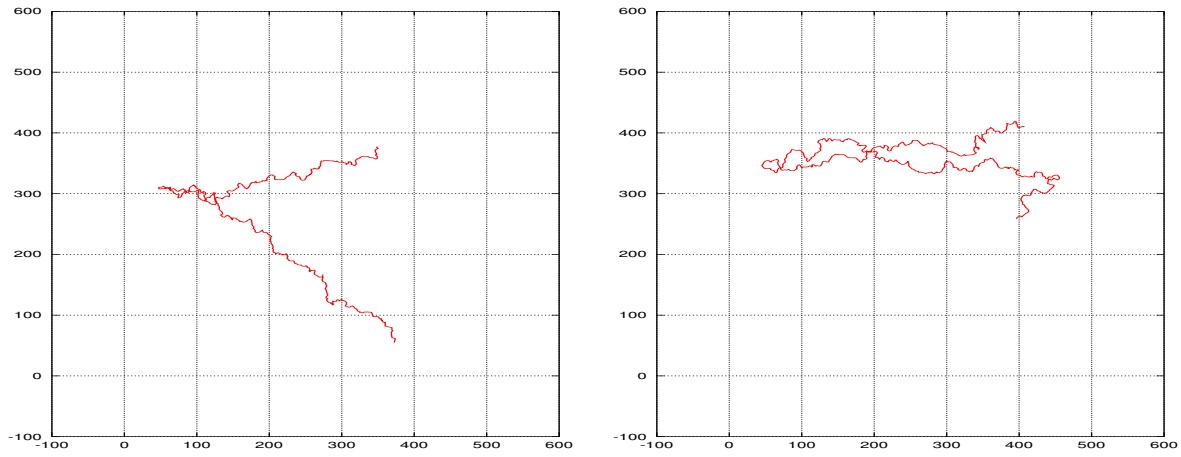
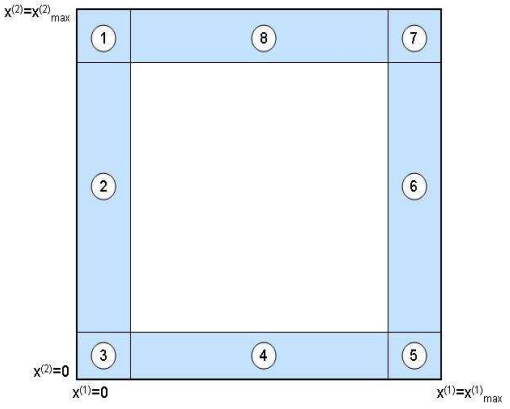


Figure 3.3: Two trajectories resulting from the Gauss-Markov Mobility Model computed with SUMI. (1 node, 500×500 simulation-area, 1000 iterations, $v_{\max} = 10$



Simulation Sub-Area	new μ_Θ Value
1	0.25 Π
2	2 Π
3	1.75 Π
4	1.5 Π
5	1.25 Π
6	Π
7	0.75 Π
8	0.5 Π

Figure 3.4: Boundary of the simulation area and corresponding changes in the μ_θ value.

in the value of μ_θ should not qualitatively influence the model's behavior outside the boundary. Figure 3.3 was computed with the improvement and the boundary area was chosen as $([0, x_{1\max}] \times [0, x_{2\max}])/([0.1x_{1\max}, 0.9x_{1\max}] \times [0.1x_{2\max}, 0.9x_{2\max}])$ (blue area in figure 3.4). The left and right trajectories show soft turns when entering the respective sub-areas of the boundary ($x_{1\max} = x_{2\max} = 500$).

Improvement 2: Speed-Modification Another drawback of the simple model is that the values of μ_v are randomly chosen and remain constant for the duration of the simulation. This does not seem to be very realistic. Although there may be systematic differences between walking speeds of elder people and younger people, the general differences in velocity distributions and ubiquitous availability of various transport alternatives in an urban area smear out the differences between the individuals so that a general average speed that does not change is not appropriate. To overcome this difficulty, a conventional discrete Markov process (Markov chain) was chosen, in order to randomly select one of several mobility states for each node in each simulation time step. Each mobility state corresponds to a general class of transport means in urban areas in the

Western World (North America and continental Europe). We distinguish between these **basic mobility states**:

- **On Foot**. In this state all forms of tool-less locomotion are included: Walking, jogging, running etc. Usage of macroscopic, public motorized tools like escalators or elevators is also included here.
- **Bike**. This state incorporates all forms of motion directly generated with the help of unmotorized mechanical devices: Bikes, Inline-Skates, scooters etc.
- **Car**. Here we include all locomotion accomplished with the help of motorized vehicles of individual traffic: Cars, motorbikes, etc.
- **Subway**. This state includes all public subway-like transportation if below earth or above except for suburban trains.
- **Suburban Train**. Includes all public suburban-train-like transportation ("S-Bahn" in Germany). Trains usually use the railway system of transnational railway.
- **Bus**. Includes all other forms of public transportation: Bus, streetcar etc.

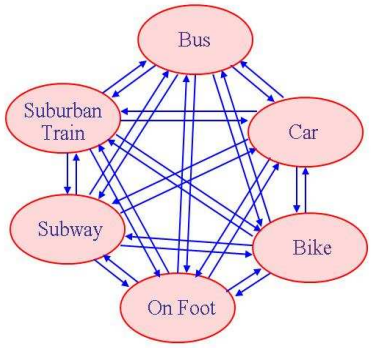


Figure 3.5: Markov Chain for Mobility State Selection.

Markov chains are computationally simple to implement and allow for an empirical determination of its parameters which are directly interpretable. Furthermore, the memorylessness of the process seems adequate for the mobility state selection because in real life, the logical choice of what other transportation means to select next or whether to remain in a mobility state depends solely on the present state and not on previous states. For a discrete Markov chain $\{X^{(t)} \mid t \in \mathbb{N}\}$ we map to every sequence of states $x = (x_0, x_1, \ldots, t = n)$ (that is every ordered sequence of values of $X^{(t)}$ at discrete points in time $t = 1, t = 2, \ldots, t = n$) to a probability

$$P(x) = P[X^{(0)} = x_0] \prod_{t=1}^{n} P[X^{(t)} = x_t | X^{(t-1)} = x_{t-1}] \tag{3.13}$$

where we have already used the memorylessness. We assume that the set of possible states is finite and thus countable. The set of possible states will be denoted by S and its number by

$s = |S|$.
Denoting $p_{ij} = P[X^{(t+1)} = x_j | X^{(t)} = x_i]$ and $q_i^{(t)} = P[X^{(t)} = x_i]$ and regarding that due to stationarity the matrix p does not depend on time, we can compute the probability for the next state from our knowledge of the present state from the lemma of total probability:

$$q_j^{(t+1)} = \sum_{i=0}^{s-1} p_{ij}\, q_i^{(t)} \tag{3.14}$$

Thus the process is completely characterized by the transition matrix $p_{ij}; 0 \leq i, j \leq s-1$. Equation (3.14) can be written in short form as $q^{(t+1)} = q^{(t)}p$. The vector q is often called a state vector for the process.
Figure 3.5 depicts the complete graph of the Markov chain of our mobility states. To determine the transition probabilities p_{ij} between the states, one needs to evaluate a broad spectrum of data which are collected by the public transportation companies (in Munich, polls are conducted in periodic time intervals by the local transportation company MVV) and by other public agencies (such as the police, traffic management agency etc.). Although the calculation of the transition probabilities from such data would be straightforward, these data are, unfortunately, not publicly available and representative own polls would have been too costly. Therefore, the transition matrix was estimated based on personal experience:

↓ From \ To →	On Foot	Car	Bike	Subway	Suburban	Bus
On Foot	0.5	0.1	0.1	0.1	0.1	0.1
Car	0.4	0.4	0.05	0.05	0.05	0.05
Bike	0.4	0.05	0.4	0.05	0.05	0.05
Subway	0.4	0.1	0.05	0.25	0.1	0.1
Suburban	0.4	0.1	0.05	0.1	0.25	0.1
Bus	0.4	0.1	0.05	0.1	0.1	0.25

Table 3.1: Transition probabilities for the mobility states

$$\Rightarrow \quad p = \begin{pmatrix} 0.5 & 0.1 & 0.1 & 0.1 & 0.1 & 0.1 \\ 0.4 & 0.4 & 0.05 & 0.05 & 0.05 & 0.05 \\ 0.4 & 0.05 & 0.4 & 0.05 & 0.05 & 0.05 \\ 0.4 & 0.1 & 0.05 & 0.25 & 0.1 & 0.1 \\ 0.4 & 0.1 & 0.05 & 0.1 & 0.25 & 0.1 \\ 0.4 & 0.1 & 0.05 & 0.1 & 0.1 & 0.25 \end{pmatrix} \tag{3.15}$$

This estimation is based on relatively large values for the probability p_{ii}: If one is in mobility state i it is likely that one will be in that state for the next time step too. Since "On Foot" (row 1) is in some sense a basic state, the probabilities to change to any other state from there are equally large. In contrast to that, people that choose motorized individual traffic vehicles ("Car", row 2) will generally have only a small tendency to change from e.g. car to public transport (park and ride) or change to "Bike". If one has traveled for a while in a car, the psychological hurdle to change to other forms of transport is very high even in case of traffic jams (inflexibility). The only significant contribution comes from the change "Car" to "On Foot" which was assumed to be quite probable since not all regions of an urban area are immediately

accessible by car due to problems in finding a parking lot etc. People in "Bike" (row 3) will have a tendency to either switch to "On Foot" (Goal is reached) or to remain in "Bike". The probability to change to public transport is low in this state due to the inconvenience of handling a bike in public transport systems. And taking the car after one has spend some time on a bike is also rather improbable. The three public transport states (rows 4,5,6) are very similar to each other, have a high probability to change to "On foot" because it can well be assumed that it is the usual case that a public transport phase is succeeded by a walk. The mixing probabilities with other forms of public transports are equally high in all three states because this concept relies on a dense, supplemental network of heterogeneous transport means.

Looking at statistical properties of this Markov chain characterized by the transition matrix (3.15), we will first state some simple properties. It is irreducible, because $\forall i, j \in S : \exists n \in \mathbb{N}$ so that $p_{ij}^n > 0$ (definition of irreducibility [177]) which follows from the strong connectivity of the graph in figure 3.5. Thus a unique stationary distribution π exists: $\pi = \pi p$. Our Markov chain is furthermore aperiodic because $\forall i \in S : p_{ii} > 0$ (which is a sufficient condition for aperiodicity [177]). Our Markov process is irreducible and aperiodic thus it is ergodic. The fundamental lemma for ergodic Markov chains [177] states that every ergodic Markov chain converges in the limit $t \to \infty$ to its (unique) stationary distribution: $\lim_{t\to\infty} q^{(t)} = \pi$. Note that this stationary distribution is independent of the start-distribution $q^{(0)}$.
Evaluating this with a computer algebra program (Matlab), we find exactly one left Eigenvalue d_1 equal to one corresponding to the left eigenvector (or more precisely the one-parametric Eigen-line) $e_1 = \lambda\ \widetilde{e_1} = \lambda\ (0.8716, 0.2646, 0.2179, 0.2023, 0.2023, 0.2023)$ with arbitrary $\lambda \in \mathbb{C}$. Matlab chooses the numeric values of $\widetilde{e_1}$ such that $||\widetilde{e_1}|| = 1$ Regarding that $\sum_{i=0}^{s-1} q \stackrel{!}{=} 1$ we set $\lambda = 1/\sum_{i=0}^{5} \widetilde{e_1}_i$ and get $e_1 = (0.4444, 0.1349, 0.1111, 0.1032, 0.1032, 0.1032)$. Letting SUMI calculate 50000 simulation iterations of the Markov chain and estimating the resulting stationary distribution by $\pi_i^{\text{estimate}} = (\text{number of occurrences of state } i)/50000$ we get $\pi_i^{\text{estimate}} = (0.44356, 0.1302, 0.11224, 0.10464, 0.10588, 0.10348)$ which is (within numerical errors) equal to e_1.
This means that in a sufficiently long simulation, a node in SUMI will roughly be 44 % of the time in state "On Foot", 13 % in state "Car", 11 % in state "Bike", and roughly 30 % of the time in one of the public transport states which is realistic on average for a person moving in an urban area.

Estimated Entity	Experiment	State 0	State 1	State 2	State 3	State 4	State 5
D_i^{estimate}	1	1.9861	1.6892	1.7160	1.2857	1.3372	1.3117
	2	1.9196	1.6761	1.6420	1.3143	1.3523	1.4000
	3	1.9563	1.8696	1.5600	1.3059	1.2597	1.3881
	4	1.9231	1.6162	1.6176	1.3538	1.4143	1.3906
	5	2.1060	1.5930	1.6721	1.4085	1.2500	1.3867
$\sigma(D_i)^{\text{estimate}}$	1	1.3726	1.1381	1.0212	0.6128	0.7251	0.5869
	2	1.3406	1.3817	1.0808	0.6446	0.8263	0.8794
	3	1.4651	1.2872	1.1689	0.5093	0.5676	0.6901
	4	1.3407	1.0797	1.0851	0.6891	0.7070	0.6025
	5	1.4632	0.8405	0.9534	0.5711	0.5809	0.7049

Table 3.2: Average stay time D_i^{estimate} in the 6 Mobility states and standard deviations $\sigma(D_i)^{\text{estimate}}$ for 5 runs a 1000 iterations with a Markov chain having the transition matrix (3.15).

Another interesting analysis is the estimation of the average time D_i that a node stays in state i. In the next section, we will see how these times can (in principle) be calculated analytically. Estimating them numerically, we let SUMI calculate 5 times 1000 iterations of the Markov chain and estimate $D_i^{\text{estimate}} = \sum$ length of connected sequences of state i with length ≥ 1 / number occurrences of connected sequences of state i with length ≥ 1. We further estimate the standard deviation of D_i as $\sigma^2(D_i)^{\text{estimate}} = \sum$(length of connected sequences of state i with length $\geq 1)^2$ / (number of occurrences of connected sequences of state i with length $\geq 1)^2 - {D_i^{\text{estimate}}}^2$. The results are displayed in table 3.2. The table shows that we can roughly classify the stay times into three classes. The longest average stay time has state "On Foot" with ≈ 2 time steps the second longest average stay time have "Car" and "Bike" with ≈ 1.7 time steps and the third class are the public transport states with ≈ 1.3. These results together with the reasonable standard deviations are realistic taken into account that for a simulation of 1000 iterations with SUMI equivalent to 3 days we have a time step duration equivalent to 4.3 minutes.

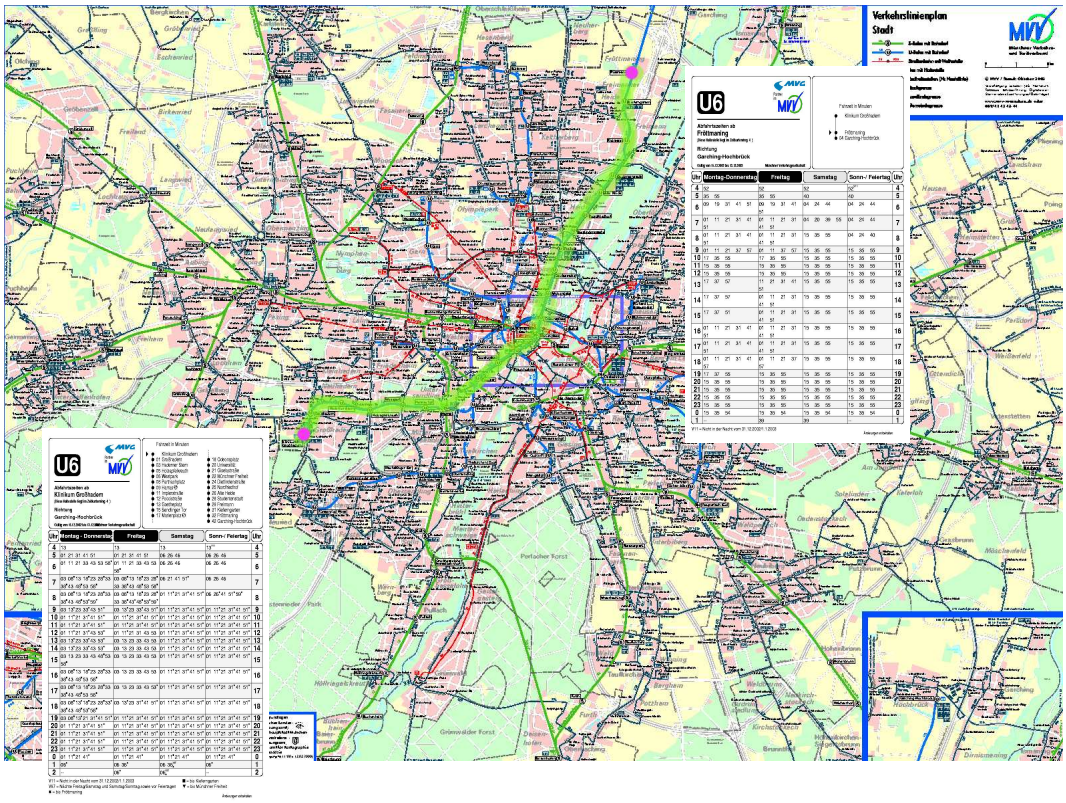


Figure 3.6: Coarse estimation of average velocity of public transport systems

In every connected sequence of mobility states, we assume a Gaussian distribution of the average velocity $\mu_v(i)$. The parameters of μ_i, σ_i of these distributions $\mathcal{N}(\mu_i, \sigma_i)$ were estimated using all available information. In case of "On Foot", "Car", and "Bike", telephone interviews with experts from Public Transport Companies, public administration, diverse scientists from the field of Traffic Management and scientific advisors of companies specialized in traffic simulation

software were conducted which yielded the results in table 3.3. In case of public transport systems, the average velocities μ_i were estimated by measuring the length of 3 suburban-, 3 subway- and 3 bus-lines on a map of the public transportation net in Munich and the durations of travel was gathered from the schedules. Figure 3.6 shows this plan together with the schedules from two stopping points for one subway line. The standard deviations were estimated on the basis of personal experience.

	μ_i (km / h)	σ_i (km / h)
On Foot	4	2
Car	20	10
Bike	15	10
Subway	35	10
Suburban	45	10
Bus	16	10

Table 3.3: Average speed μ_i and estimated standard deviation σ_i in the mobility states i (Parameters of Gaussian distribution $\mathcal{N}$).

3.7 Resting Times

The Gauss Markov Model does not include systematic means for pause- or resting times. In SUMI two mechanisms for resting times for the nodes (entities whose movements are simulated (e.g. individual persons)) were integrated: random resting times and periodic resting times.

Random Resting Times Moving in an urban area implies times of resting. Short times include episodes of standing in front of shop windows or sitting on a park bench for a little rest. Longer times include drinking a coffee with a friend just met or waiting in a doctor's practice. Since it would be costly to model all those cases separately, SUMI uses again a simple Markov chain to model the sequence of states of resting and moving. The chain is depicted in figure 3.7. It has two states: A resting state "r" and a moving state "m".

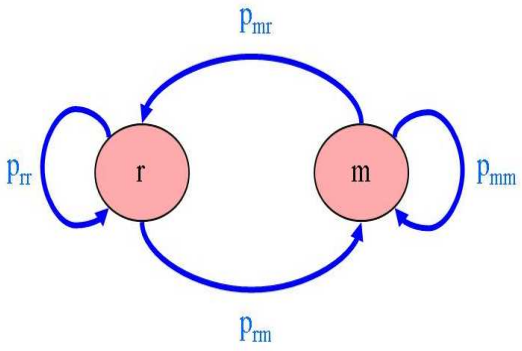


Figure 3.7: Markov Chain for Resting and Moving.

More desirable as input parameters than the four transition probabilities are the average duration $\bar{d}_r$ of a resting period for a node and the average number of such resting periods $\bar{z}$ if we simulate n time steps. It is straightforward to compute the transition probabilities if $\bar{d}_r$ and $\bar{z}$ are given as we will see now:

Let the random variable

$$T_{ij} = \min\{n | n \geq 1 \wedge (X_n = j \text{ if } X_0 = i)\} \tag{3.16}$$

count the number of steps that it takes the Markov chain to travel the way from state i to state j (or more precisely: The number of transitions starting from state i until state j is reached for the first time). T_{ij} is called **hitting time** [177].
Furthermore we denote for the expectation value of T_{ij}

$$h_{ij} = E(T_{ij}). \tag{3.17}$$

We can then easily justify that [177]

$$\begin{aligned} h_{ij} &= E(T_{ij}) = \sum_{k \in S} E[T_{ij}|X^{(1)} = k]\ P[X^{(1)} = k] \overset{(\star)}{=} \sum_{k \in S} E[T_{ij}|X^{(1)} = k]\ p_{ik} \\ &\overset{(\star\star)}{=} p_{ij} + \sum_{k \neq j} E[T_{ij}|X^{(1)} = k]\ p_{ik} \overset{(\star\star\star)}{=} p_{ij} + \sum_{k \neq j} (1 + E[T_{kj}])\ p_{ik} \\ &\overset{(\star\star\star\star)}{=} 1 + \sum_{k \neq j} h_{kj}\ p_{ik}. \end{aligned} \tag{3.18}$$

because we have

$(\star)$ $P[X^{(0)} = i] = 1$ (Assumption in Def. (3.16))

$(\star\star)$ $E[T_{ij}|X^{(1)} = j] = 1$ (Def. (3.16))

$(\star\star\star)$ $\forall k \neq j:\ E[T_{ij}|X^{(1)} = k] = (1 + E[T_{kj}])$ (Because of memorylessness)

$(\star\star\star\star)$ $\sum_{k \in S} p_{ik} = 1.$

Equation (3.18) is a system of linear equations for h_{ij} which (regarding that $p_{rr} = 1 - p_{rm}$ and $p_{mm} = 1 - p_{mr}$) reads in the case of our simple chain of figure 3.7:

$$h_{rr} = 1 + p_{rm} h_{mr} \tag{3.19}$$

$$h_{mm} = 1 + p_{mr} h_{rm} \tag{3.20}$$

$$h_{rm} = 1 + h_{rm} - p_{rm} h_{rm} \tag{3.21}$$

$$h_{mr} = 1 + h_{mr} - p_{mr} h_{mr} \tag{3.22}$$

which has the simple solution

$$h_{rr} = 1 + p_{rm}/p_{mr} \tag{3.23}$$

$$h_{mm} = 1 + p_{mr}/p_{rm} \tag{3.24}$$

$$h_{rm} = 1/p_{rm} \tag{3.25}$$

$$h_{mr} = 1/p_{mr} \tag{3.26}$$

Since h_{rm} is equal to the average duration of a connected resting period:

$$h_{rm} = \bar{d}_r \tag{3.27}$$

and h_{mr} is equal to the average duration of a connected moving period, the average number of waiting intervals per n simulation steps is

$$\bar{z} = \frac{n}{h_{rm} + h_{mr}} \quad . \tag{3.28}$$

So we finally get

$$p_{rm} = \frac{1}{\bar{d}_r} \quad \text{and} \quad p_{mr} = \frac{1}{n/z - \bar{d}_r} \tag{3.29}$$

So parameters can be chosen according to the desired specific simulation characteristics.

Periodic Resting Times Apart from random resting times, people also have periodic occupations which make them stay in one place. Among these occupations are sleeping and certain types of work. So SUMI provides a means for computing these periodic resting times.
In SUMI, the periodic resting times are computed before the actual simulation starts in form of a schedule. This is also the case with the mobility states of the Gauss-Markov-Model Improvement and the random resting times. Such a schedule in principle takes the form of a $n_{\mathcal{X}} \times n_{\text{Iterations}}$ Matrix where $n_{\mathcal{X}}$ is the number of nodes (entities whose movements are simulated) and $n_{\text{Iterations}}$ is the number of iterations. The entries of the schedule matrix are either numbers of mobility states or symbols "w" (wait) and "m" (move) in case of random resting times and "p" (periodic resting) and "m" (move) in case of periodic resting times.
For the periodic resting time intervals, we again have to specify the average duration of these intervals $\bar{z}^{(p)}$ and the average number of these intervals $\bar{d}_r^{(p)}$ during the simulation. The default values are chosen so that we have two periodic resting periods per simulated day and the average duration of that period is the number of time steps equivalent to 5.0 hours.
For every node the periodic resting schedule is computed as follows:

- Input: average length of periodic resting intervals $\bar{d}_r^{(p)}$ and average number of these intervals $\bar{z}^{(p)}$.
- Compute the length of the basic partition $a = n_{\text{Iterations}}/\bar{z}^{(p)}$ of the schedule.
- Compute an random initial offset (Gaussian distribution) $a' \sim \mathcal{N}(0.5a, 0.2a)$.
- Compute the resting interval medians $m_i = a' + i\ a$.
- Compute Gaussian distributed deviations for the interval Medians: $m_i = m_i + \Delta m_i$ with $\Delta m_i \sim \mathcal{N}^{(2)}(0, 0.1a)$
- Compute Gaussian distributed deviations for the interval lengths: ${d_r^{(p)}}_i = {d_r^{(p)}}_i + \Delta {d_r^{(p)}}_i$ with $\Delta {d_r^{(p)}}_i \sim \mathcal{N}(0, 0.2a)$

In case of the random resting periods, the nodes remain at the position they have reached when the resting interval begins. This behavior is not realistic in case of the periodic resting periods, since these periods usually take place at a fixed location (home or working place). Therefore, for

every node, two random resting locations $x_r^{(1)}$ and $x_r^{(2)}$ are determined: $x_r^{(1,2)} \sim \mathcal{R}^2(a,b)$ where $[a_1, b_1]$ is the x-extension of the simulation area and $[a_2, b_2]$ is the y-extension of the simulation area. When the scheduled periodic resting period begins, the node moves to one of the two locations and stays there until the periodic resting period is over.
In order to determine whether a node has reached its pre-computed resting position, we make use of the pointReachTolerance technique described in paragraph 3.8 in figure 3.10. In order to achieve a more realistic model the node makes small random movements (controlled by a parameter γ) around the resting position x_r:

$$\begin{aligned} x &\sim \mathcal{R}^2[x_r - (\gamma,\gamma), x_r + (\gamma,\gamma)] \\ v &\sim \mathcal{R}[0,1] \\ \theta &\sim \mathcal{R}[0,2\pi] \end{aligned} \tag{3.30}$$

The movement to the resting locations occurs in a straight line from the location where the node is when the periodic resting period begins. The direction value θ is adapted to the direction towards the resting point and kept fixed. The speed is accelerated to the maximum speed which is taken as the maximum over the average speeds in the mobility states $v_{\max} = \max\{\mu_0, \mu_1, \ldots, \mu_{s-1}\}$:

$$v^{(n)} = 0.9v^{(n-1)} + 0.1v_{\max} \tag{3.31}$$

The choice for such a rather unrealistic movement pattern towards the resting location was made because otherwise it would be very hard to control the time that a node actually rests in its resting location. Allowing a certain share of the resting period to be used for traveling to the resting location, we must ensure that a sufficient share of the resting periods can be used for resting. In order to estimate this, we have to calculate the average distance $||x - x_r||$ between the position reached x when the period begins to the resting location x_r. If we denote the components of x with A and B and the components of x_r with C and D and assume $A, C \sim \mathcal{R}(a_1, b_1)$ and $B, D \sim \mathcal{R}(a_2, b_2)$ we have to compute the expectation value of the Random Variable $\rho = ((A-C)^2 + (B-D)^2)^{\frac{1}{2}}$:

$$E(\rho) = \int_{a_1}^{b_1}\int_{a_2}^{b_2}\int_{a_1}^{b_1}\int_{a_2}^{b_2} ((a-c)^2 + (b-d)^2)^{\frac{1}{2}} \frac{1}{(b_1-a_1)^2}\frac{1}{(b_2-a_2)^2}\,\mathrm{d}a\,\mathrm{d}b\,\mathrm{d}c\,\mathrm{d}d \tag{3.32}$$

While there is an analytical solution for this integral [56] it contains too many case differentiations to be computationally pleasant. Therefore this expectation value is determined numerically by SUMI.

On the basis of this calculation and on a given parameter for the allowed share of the periodic resting period that can be used for traveling to the resting location and assuming an average speed for this trip of $1/2\ v_{\max}$, we can estimate whether the value of $\vec{d}_r^{(p)}$ needs to be enlarged in order to guarantee a sufficiently large resting time.

3.8 Group Mobility Model

One of the major challenges in choosing an appropriate group mobility model is to integrate group motion and individual motion which is governed by the adapted model described in

the previous sections. When discussing the approach taken in SUMI, we will therefore have to describe the actual group mobility model in parallel to describing its integration with the individual motion.
A very promising approach to achieve a seamless integration of individual and group motion without pre-computed schedules that is worth trying is to use equations of motion with multiple charges and a classical field theory. This physics motivated approach would first determine the number of groups desired and would use one color of "charge" per group. In individual motion, these charges are "turned off" and any individual mobility model can govern the motion of the nodes. If group motion for one or more groups is desired, the specific group charge is turned on and the nodes are attracted according to a force which corresponds to a certain potential. The potential for a classical electric field would be $\phi(x) = C\ 1/|x|^2$, which might not be strong enough to ensure that the nodes are really attracted far enough towards each other, because as it is known from Kepler's solution to the two-body problem with gravitational potential which is also reverse quadratic that elliptic rotation around a center of mass, hyperbolic or parabolic scattering motion can arise which is not desired for our application. Thus a potential with reverse cubic form might have to be chosen. The potential could be complemented by a repulsive potential for short distances. This short range potential would ensure that the nodes are not attracted to a single point but keep a certain distance from each other. Assuming unit mass for the nodes, the Hamiltonian for a single group of k nodes would be

$$H(p_1, p_2, \ldots, p_n, x_1, x_2, \ldots, x_n) = \frac{1}{2}\sum_{i=1}^{n} p_i^2 \ + \ \frac{1}{2} q^{\text{group}} \sum_{i \neq j}^{n} \phi(||x_i - x_j||). \tag{3.33}$$

The approach was dropped mainly because the parameters of this model are too hard to interpret and the effect of changing one of them (e.g. the mass of the nodes) is too indirect and not very demonstrative. But nevertheless the method is worth trying, especially if the computation of group schedules is not possible or not desired.

The approach taken in SUMI uses pre-computed group motion schedules. This means that the periods when certain sets of groups move as a group are computed before the actual simulation starts based on a stochastic model.

Schedule Generation As in the case of resting times, we assume that there are random group times and periodic group times. Random group times arise, when several people e.g. make a one-time appointment for a spare time activity or simply meet by chance. Periodic group times occur when several people meet on a regular basis e.g. in a sports club for team sport practice or at work. Another distinction that has to be made is whether the group in question is a group that stays in one place after it has formed ("resting group") or whether it is a group that moves after it has formed ("moving group"). An example for a resting group is a basketball team practice (with small location deviations constrained by the gym's dimensions). An example for a moving group is a group of joggers.

For the computation of periodic group times schedule we proceed as follows:

- Input: average length of periodic group intervals $\bar{d}_r^{(g,p)}$, average number of these intervals $\bar{z}^{(g,p)}$ and average number of group members $\bar{g}_r^{(g,p)}$.
- On the $n_{\mathcal{X}} \times n_{\text{Iterations}}$ schedule matrix, deterministically compute a grid with mesh-width $n_{\text{Iterations}}/\bar{z}^{(g,p)}$ and mesh-height $\bar{g}_r^{(g,p)}$.

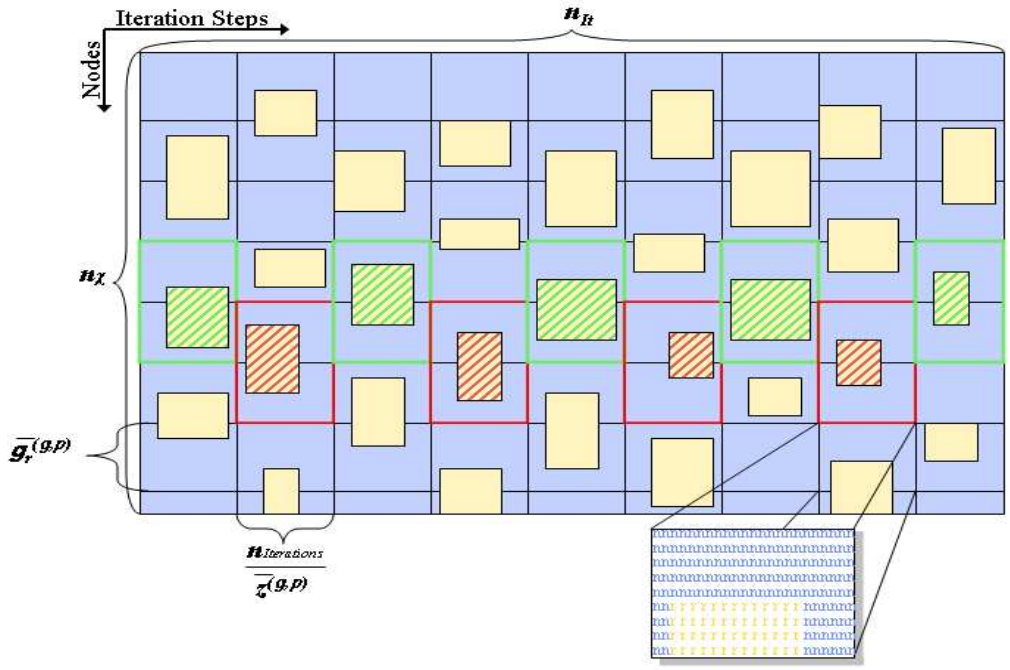


Figure 3.8: The structure of a periodic group schedule. The schedule is structurally equivalent to a $n_{\mathcal{X}} \times n_{\text{Iterations}}$ matrix of symbols. If the symbol in the schedule is an "n" there is no group motion. If the symbol is an "r" or an "m" we have a resting group or moving group respectively. For the explanation of the grid dependence of the group-rectangles see text.

- Iterate over the grid cells:
  - If the vertical cell index and the horizontal cell index are even or if both indices are odd, consider this cell and the cell immediately below. Inscribe a rectangle of average width $\bar{d}_r^{(g,p)}$ and average height $\bar{g}_r^{(g,p)}$ into this double cell in the following way:
  - Let the max. allowed deviation (horizontally and vertically) be δ; compute horizontal deviation $\Delta\bar{d}_r^{(g,p)} \sim \mathcal{R}(-\delta\bar{d}_r^{(g,p)},\ \delta\bar{d}_r^{(g,p)})$ and vertical deviation $\Delta\bar{g}_r^{(g,p)} \sim \mathcal{R}(-\delta\bar{g}_r^{(g,p)},\ \delta\bar{g}_r^{(g,p)})$.
  - For every "row" of such cells determine randomly(Random-Variable A) if its a resting group $A = r$ or a moving group $A = m$: $A \sim B(p_{\text{move}})$ (Bernoulli).
  - For every "row" of such cells determine randomly a center $(m^{(p)}_{\text{horizontal}}, m^{(p)}_{\text{vertical}})$ for the inscribed rectangle: $m^{(p)}_{\text{horizontal}} \sim \mathcal{R}((\delta + \frac{1}{2})\bar{d}_r^{(g,p)}, n_{\text{Iterations}}/\bar{z}^{(g,p)} - (\delta + \frac{1}{2})\bar{d}_r^{(g,p)})$ and $m^{(p)}_{\text{vertical}} \sim \mathcal{R}((\delta + \frac{1}{2})\bar{g}_r^{(g,p)}, \bar{g}_r^{(g,p)} - (\delta + \frac{1}{2})\bar{g}_r^{(g,p)})$. The complicated expressions are due to the fact, that a rectangle with sufficient width and height needs to be inscribable around this center.
  - Compute the inscribed rectangle with width $\bar{d}_r^{(g,p)} + \Delta\bar{d}_r^{(g,p)}$ and height $\bar{g}_r^{(g,p)} + \Delta\bar{g}_r^{(g,p)}$ symmetrically around the center.

The mathematical details of this procedure are not that important. More important is the interpretation of the computed schedule. Each computed rectangle in the schedule corresponds to one group motion interval. The vertical extension of the rectangle determines the nodes

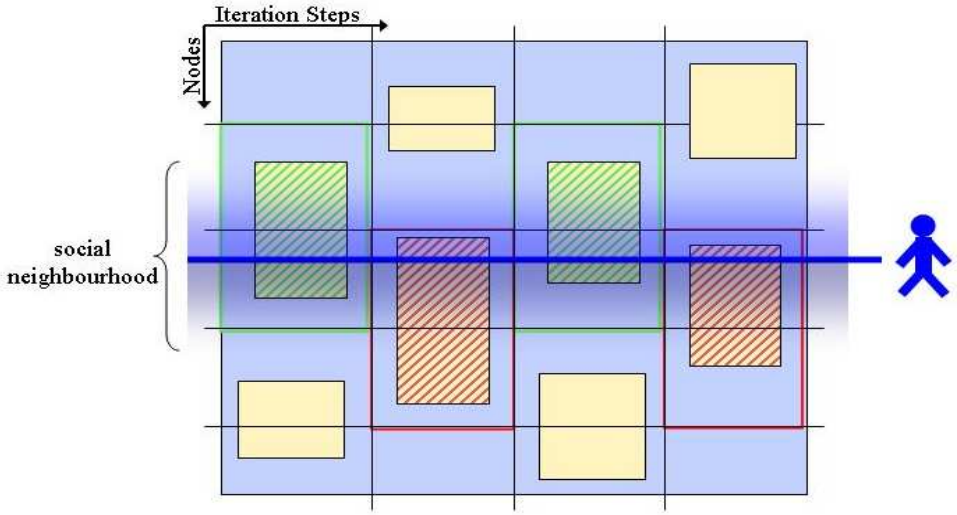


Figure 3.9: The social neighborhood of a node (person).

which take part in the group motion and the horizontal extension determines the length of the group interval. Figure 3.8 shows what such a schedule looks like in principle. The blue area in the figure corresponds to individual motion and the pale yellow rectangles correspond to group motion. Below the actual schedule, one cell is depicted in more detail which shows the symbols "n" for no group motion and "r" for resting group. In order to illustrate what is meant with "rows" of rectangles with common center location within the corresponding double cell and common character (resting or moving periodic group), figure 3.8 shows such a row by a green stripe pattern and the corresponding double grid cells with a green frame. Shown with red stripe pattern and with their corresponding double grid cells with a red frame is another row of group-time rectangles. Each pair of such rows has a comb like displacement with respect to each other. In that way, some nodes are involved with two periodic groups (rows) and some nodes are involved in only one periodic group or none at all. Figure 3.9 is intended to illuminate this construction principle in more detail. The person which corresponds to the node whose row index is depicted by the blue line is part of two periodic groups: the group with green stripes and the group with red stripes. The varying vertical extensions of the group motion rectangles model the fact that some people take part in the group more regularly (those at the center of the vertical extension) and some less regularly (those towards the vertical edges of the rectangle). The varying horizontal extensions model the fact that not all group motion intervals have exactly the same length.

By using only vertically compact geometrical structures, we assume that the nodes (persons) have been sorted in a way that reflects their social affinities. With this geometry of the schedule, it is not possible for a node that is vertically in the center of a periodic group "row" not to take part the group motion. Thus we have several typical types of persons: selective but loyal social beings (membership in one periodic group and reliable participation), non-commited yet socially interested persons (membership in two periodic groups and unreliable participation; e.g. the person in figure 3.9) and some "loner"-type persons (membership in one periodic group and unreliable participation). With this type of sorting, the social "neighborhood" of a person is a fuzzy vertical interval around its own vertical index as depicted in figure 3.9. These are the

people that this person has a social affinity to.

All that has been said applies to random groups as well. The schedule for the resting groups is computed first. Then a specified number of random group intervals (rectangles) are added to the schedule in exactly the same way as the group intervals for the periodic groups except that the interval (rectangle) medians $(m_{\text{horizontal}}, m_{\text{vertical}})$ are randomly chosen within the whole $n_{\mathcal{X}} \times n_{\text{Iterations}}$ schedule matrix: $m_{\text{horizontal}} \sim \mathcal{R}(0, n_{\text{Iterations}})$ and $m_{\text{vertical}} \sim \mathcal{R}(0, n_{\mathcal{X}})$. If the so computed rectangle overlaps with another rectangle, it is discarded and a new rectangle is computed.

In section 2.3.1 it has been discussed what we understand by an **abstract existence of a group** and that **Ad-Hoc-Groups** are defined as instantiations of an abstract group. Applying these notions to our simulation we find that e.g. the rectangles with a green stripe pattern in the corresponding double grid cells with a green frame in figure 3.8 are all Ad-Hoc-Group manifestations of a single abstract group. This abstract group may again be the aforementioned basketball team which meets for practice in regular time intervals but not all members of the team will participate in all practices. So each the group of every practice is a possibly different Ad-Hoc Group manifestation. For calculations which will be discussed in chapter 5 we can assign an index to every Ad-Hoc Group (every rectangle in the schedule) as well as an index for every abstract group. The abstract group will be indexed with the vertical index of the corresponding double grid cell.

Group Motion The actual group motion has two phases. The first phase is the phase of group formation. Here we use the same mechanism as described in the section about periodic resting times: A center for the group (a meeting point) is randomly computed within the simulation area. When the group motion interval starts, all group members move in a straight line towards this meeting point. As in the case of periodic resting intervals, the plausibility of the group interval length is checked to ensure that a sufficient share of the group motion interval is reserved for the group to move or to rest together and that the group motion interval is not used entirely for the journey to the meeting point. After all members have arrived at the meeting point, the group motion (or resting) starts.
In order to detect that a node has reached a given point, we define a velocity dependent point reach tolerance for every node. Using a fixed criterion leads to ugly oscillations around the meeting point. This behavior occurs if a rather strict fixed criterion $||x - x_{\text{meetingPoint}}|| < \delta$ occurs together with a high velocity. If the meeting point criterion is not met altough the node is already comparatively close, the next iteration will make the node overshoot the mark. Since the movement is directed along the straight line connecting the recent location with the meeting point, the node will be forced to make a 180 degree turn and the next iteration will take the node back across the meeting point again and so forth. Therefore the point reach tolerance needs to be velocity dependent. In one iteration of duration Δt a node will travel a distance of $v(t)\Delta t$. Since we have $\Delta t = 1$, the optimal point reach tolerance must be equal to $1/2\ v(t)$ so that no oscillations occur.

If it is a resting group, and if all members of the group have reached the meeting point, all members will rest there until the group period is over. If its a moving group and if not all members of the group have reached the meeting point, the nodes which have reached the meeting point will rest there and wait for the others to arrive. For any case of resting or waiting nodes we apply a more realistic model than just to fix the locations and set the velocity to zero. Instead we allow for small random movements around the meeting point $x_{\text{meetingPoint}}$ controlled

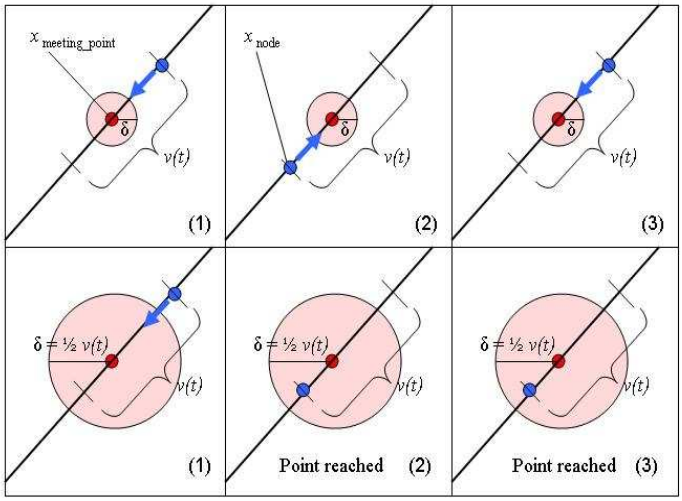


Figure 3.10: Point reach tolerance and oscillations. Upper row: if the point reach tolerance δ is fixed and too small, an oscillation will occur. Lower row: if $\delta = \delta(v(t))$ is velocity dependent, no oscillations can occur.

by a parameter γ:

$$\begin{aligned} x &\sim \mathcal{R}^2[x_r - (\gamma,\gamma), x_r + (\gamma,\gamma)] \\ v &\sim \mathcal{R}[0,1] \\ \theta &\sim \mathcal{R}[0,2\pi] \end{aligned} \tag{3.34}$$

This model has also been applied in case of individual resting times (see equation 3.30).

If its a moving group and if all members of the group have reached the meeting point, group motion will start. The group mobility model we use is a variant of RPGM or Nomadic Group Mobility Model [22] respectively. The group has a logical center (reference point) which moves according to the modified Gauss-Markov-Model of individual motion that has been described in previous sections. The group members follow the reference point with a random deviation of $\Delta x_{1/2} \sim \mathcal{R}(-\delta_{\text{interGroupTolerance}} x_{1/2}^{\text{group}}, \delta_{\text{interGroupTolerance}} x_{1/2}^{\text{group}})$ and an analogous random speed and direction deviation which are all controlled by the inter group movement tolerance parameter $\delta_{\text{interGroupTolerance}}$:

$$x_{1/2}^{\text{member}} = x_{1/2}^{\text{group}} + \Delta x_{1/2}^{\text{group}} \tag{3.35}$$

$$v^{\text{member}} = v^{\text{group}} + \Delta v^{\text{group}} \tag{3.36}$$

$$\theta^{\text{member}} = \theta^{\text{group}} + \Delta\theta^{\text{group}} \tag{3.37}$$

For the parameter γ from above we also use the inter group movement tolerance $\gamma = \delta_{\text{interGroupTolerance}}$. This parameter is chosen according to the social theories of propinquity effects introduced in chapter 2 in section 2.1.1.1. We set $\delta_{\text{interGroupTolerance}}$ to the simulation's equivalent of 30 meters.

For reasons of computational efficiency, we compute the group's reference point movements (locations, velocities and direction) for all group motion intervals before the actual simulation.

Thus, instead of symbols, the group schedule contains only hash values that reference a group motion interval object, which contains all data about the group (members, resting or moving, meeting point etc.) and its motion patterns during that interval. The patterns are computed for the whole length of the interval, neglecting the journey times of the member nodes to the group meeting points, which is an acceptable overhead. These objects are readily transformable into structs in case a C porting is necessary.

3.9 SUMI

The SUMI application assumes the following precendence order: First, the group schedule is evaluated (random and periodic) then the resting schedule is evaluated (random and periodic) and if both schedules show individual motion then the node is propagated according to the modified individual Gauss-Markov model.

The color model that SUMI uses for coloring the nodes in its simulation visualization is optimized for maximum dispersion across the spectrum of available colors. In an RBG color space, we achieve maximum dispersion, if we place the (r, g, b) color vectors of the nodes on a unit sphere so that their minimal angles are maximal. The problem of placing n points on the surface of an m-dimensional unit sphere so that the minimal angle is maximized is well known in mathematics (see [119, 170, 193]), and is non trivial. For 3 dimensions, [170] provides pre-computed solutions for $n = 1, 2, \ldots 100$ which are used to color the nodes. If more than 100 nodes are to be visualized, colors are used multiple times.

Figures 3.11 and 3.12 show intermediate screenshots for a SUMI simulation run with 5 nodes. The red and blue lines indicate the innermost extension of the tolerance area depicted in figure 3.4. The upper left subfigure of figure 3.11 shows the five nodes with their trajectories after a few steps.
The upper right subfigure shows the state of the simulation after a few more steps. A periodic resting period is scheduled for the red node and it is on its way to the periodic resting point which is shown as a black cross in the lower left corner of the simulation area. The trajectory is a straight line from the point that the red node had reached when the periodic resting interval began to the resting point.
In the lower left subfigure, the red node has reached its resting point and stays there. In the meantime, a moving group period has started with the violet and the dark grey nodes. They are both already on their way to the group's meeting point which is depicted with a grey cross. in the lower right subfigure of figure 3.11, both nodes have just reached the group's meeting point and group motion is about to begin.
The left figure of figure 3.12 shows the situation after a few more steps: While the red node is still resting, the violet and the dark grey node move together as a group. The group's reference point is shown with the help of a grey hourglass symbol.
Some steps later we have the picture of the right subfigure. The green node went on its way to a resting place and is staying there. The resting period of the red node has ended and it has continued its motion. The group is still moving together.

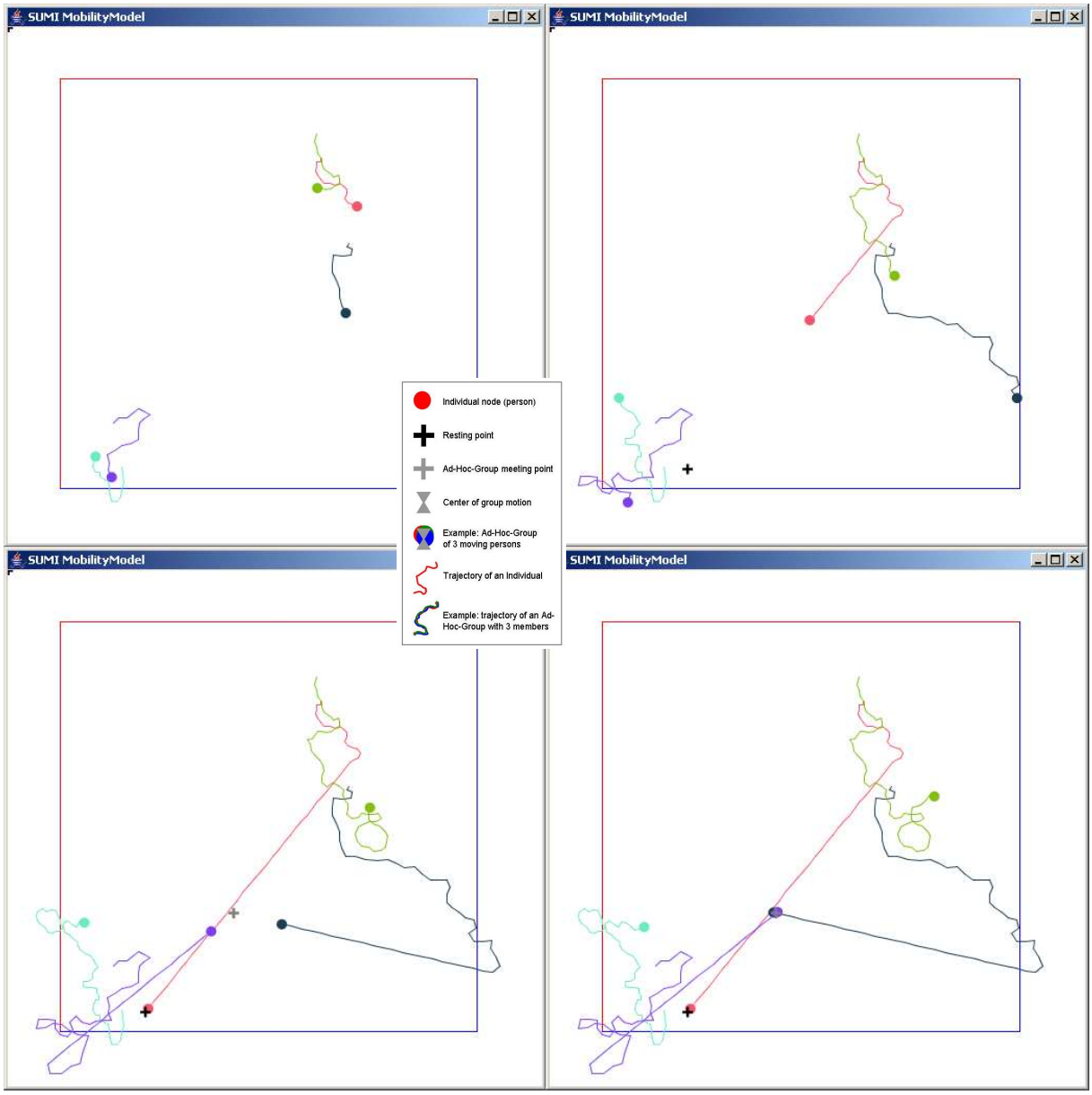


Figure 3.11: SUMI simulation with 5 nodes; $\alpha = 0.75$, $n_{\text{iterations}} = 1000$

3.10 Similarity Measure

For the $\mathbb{R}^n$, the simple distance metric $d(x, y) = ||x - y||$ resulting from the ordinary Euclidean norm $||x|| = (x_1^2, x_2^2, \ldots, x_n^2)^{1/2}$ is sufficient to compare two velocity / location vectors with the goal of finding clusters or groups of nodes. The distance metric can easily be transformed in a standard way [161] into a similarity measure via

$$\text{sim}(x, y) = \frac{1}{1 + d(x, y)} \tag{3.38}$$

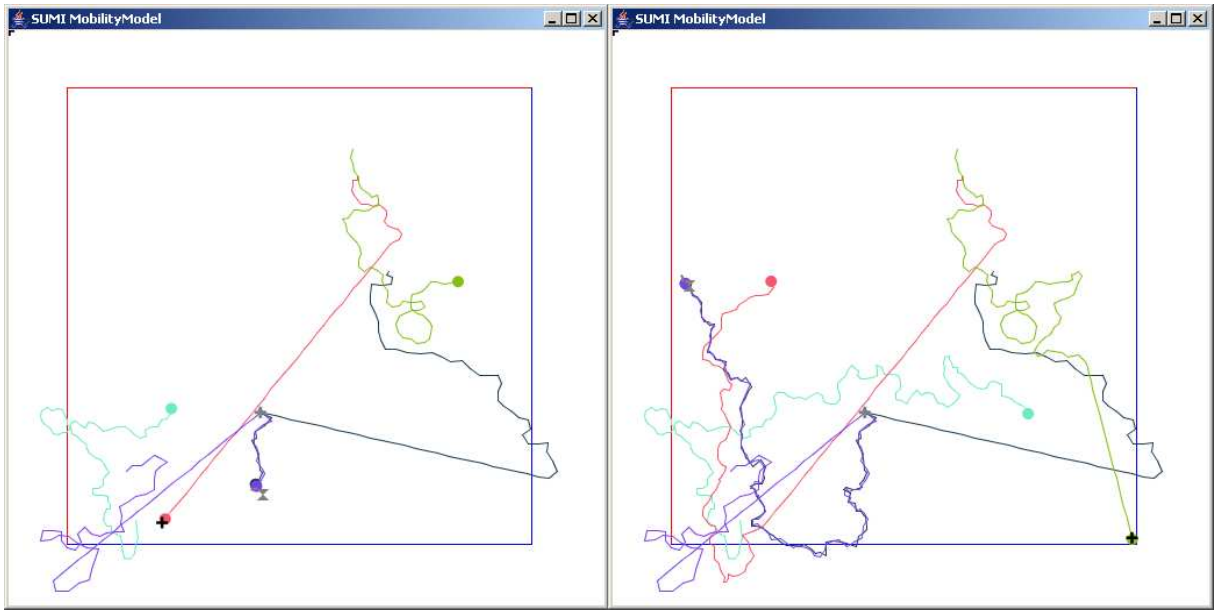


Figure 3.12: SUMI simulation with 5 nodes; $\alpha = 0.75$, $n_{\text{iterations}} = 1000$

A reasonable notion of similarity always requires a precisely defined field of application. Our ultimate goal in defining a similarity measure on a subset of the data in a community's information- and knowledge-space (such as the location and velocity of community members) is to use this similarity measure for the detection and characterization of Ad-Hoc-Groups. This topic will be dealt with in more detail in chapter 5. However, in terms of detecting such socially relevant clusters in human spatial movement in 2 dimensions on the basis of a similarity measure like (3.38), some observations concerning the similarity measure can be made.
The **first observation** is that the similarity of the velocity vectors of two nodes should generally have the same significance as the similarity between their location vectors. If two persons have very similar locations but their velocity vectors are very dissimilar (e.g. have opposite direction) these persons have a likewise small similarity with respect to each other as in case of similar velocity vectors and dissimilar locations. This point of view applies to a wide variety of applications dealing with similarities of persons with respect to location and velocity. A simple way of respecting this equality in the roles of location and velocities is to define the distance metric over the space $\mathbb{R}^2_{\text{location}} \times \mathbb{R}^2_{\text{velocity}} = \mathbb{R}^4$. This corresponds to treating location and velocity "on an equal footing". Although the velocity $v(t) = \dot{x}(t)$ can normally simply be computed from a trajectory $x(t)$ through differentiation, we will keep both quantities in our formalism on an equal footing, which allows us to easily compare two motion snapshots (with location and velocity) with one another through a similarity measure. Furthermore this bookkeeping allows us to represent cases more easily where the technology for location measurement only allows for a time-discretized detection and representation of the location and the velocity is detected independently which can produce slight inconsistencies (see figure 3.13).

At this point, some remarks about wording have to be made: In physics, classical Hamiltonian mechanics is defined over a space of locations x and conjugated impulses p (under simple conditions we have $p = m\dot{x} = mv$) which has a symplectic metric and is usually called a Phase Space. The mathematical properties of such a phase space are chosen in a way that motion patterns

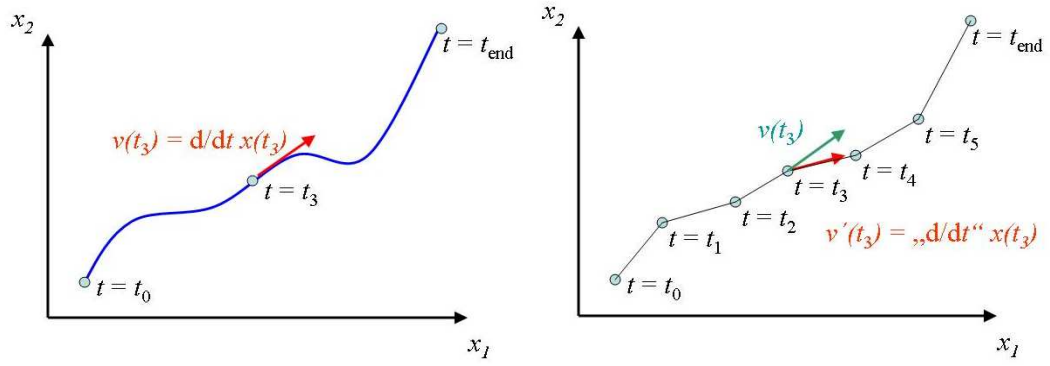


Figure 3.13: Inconsistencies through discretization. The left figure shows a 2D-trajectory $x(t)$ and the velocity vector $v(t_3)$ gained through differentiation in a point $x(t_3)$. The left subfigure shows a discretized version of the same trajectory. In the same point ($x(t_3)$), the velocity vector from the smooth curve is depicted in green ($v(t_3)$). Estimating the velocity in $x(t_3 + \mathrm{d}t)$ would yield $v'(t_3)$.

governed by the laws of physics can be conveniently described. Thus, although we will use the usual Euclidean norm instead of a symplectic norm, we will call our space $\mathcal{L} \subseteq \mathbb{R}^4$ a **Location Phase Space** in which the spatial movement that is observed or simulated can be conveniently described for our purposes.

Before we can discuss the **second observation** it is necessary, in anticipation of the discussion of chapter 5, that the upper formulation "socially relevant clusters of persons with respect to spatial movement" is further refined and substantiated.
Picking back up the discussion of chapter 2, we call a point in the Location Phase Space together with a point in time a **spatio-temporal situation** $s \in \mathcal{L} \times T$ and a simply connected subset $S \subset \mathcal{L} \times T$ a **generalized spatio-temporal situation**. A person is said to be **involved** in a generalized spatio-temporal situation S iff $\forall t \in T\ (x_t^{\text{person}}, v_t^{\text{person}}, t) \in S$. One possibility to define that a cluster of persons is a socially relevant cluster of persons with respect to spatial movement is to say that given a generalized spatial situation $S \subsetneq \mathcal{L} \times T$

- all the persons in the cluster are involved in S
- $\forall s \in S$ a social relation exists on that cluster of persons whose graph is strongly connected.

For example a team of soccer players on the soccerfield is "socially relevant with respect to spatial movement" in a generalized spatial situation which is isomorphic to $[0, 100]_{(\text{meter})} \times [0, 50]_{(\text{meter})} \times [-3.5, 3.5]_{(\text{meter/second})} \times [-3.5, 3.5]_{(\text{meter/second})} \times [0, 90]_{(\text{minute})}$ which is the formalization of a 90 minute football game on a usually sized pitch with maximum velocity of about 10 m/s. The soccer team is completely connected in this generalized spatial situation by a "plays-soccer-with" social relation and probably by many other relations such as "talks-to" "knows personally" etc.. Regard that the "knows personally" relation does not contribute to the team of soccer players being "socially relevant with respect to spatial movement" because this relation is not spatially dependent. It exists for all possible spatial situations $s \in \mathcal{L} \times T$. In chapter 5 we will discuss techniques to conclude for a spatial cluster of persons whether it is "relevant with respect to spatial movement" and from that to conclude whether it is "socially relevant" per se (this notion includes social relations without spatial dependence).
Since the variety of social relations R is manifold, we could, for example, render our notion more

precise by saying that a cluster of persons is socially relevant in a generalized spatial situation S with respect to social relation R if $\forall s \in S$ the graph of R is strongly connected.
The **second observation** is closely related to the discussion of "social relevance with respect to spatial movement": The goal is to define a similarity measure that can be used to cluster persons with respect to spatial movement so that the resulting clusters have a high probability of being "socially relevant with respect to spatial movement". In terms of a social relation with spatial dependence, we find that the similarity measure should generally depend on the relation in question: $\mathrm{sim}(x, y) = \mathrm{sim}_R(x, y)$. In the upper example of the football players the similarity measure with respect to the "plays-soccer-with" relation should strongly decrease or even have a "cutoff" when $d(x, y) > 120$, because if the distance between two persons is larger, it is very improbable that they are playing soccer with one another. On the other hand, people standing in line at a fast food restaurant counter will have a very small distance to each other but will in general not represent a cluster that is "socially relevant with respect to spatial movement". Because an adapted similarity measure for all types of relations cannot be defined for obvious reasons, a compromise must be found that covers most typical cases or that at least excludes some very improbable ones. Since a social relation that is spatially dependent requires a minimum distance a (communication distance, range of sight etc.) we will combine the similarity measure (3.38) with a generic "cut-off" function:

$$\mathrm{sim}(x, y) = \begin{cases} 1/(1 + d(x, y)) & \text{if} \quad d(x, y) < a \\ 1/(1 + d(x, y))\ \mathrm{e}^{-d(x,y)} & \text{if} \quad d(x, y) \geq a \end{cases} \tag{3.39}$$

Chapter 5 will further elaborate on questions concerning the spatial similarity measure and related heuristics.

Summary

Data for individual user modeling can be distinguished as either explicitly or implicitly collected. Collaborative information- or Knowledge Spaces (CIKS) in communities broaden the spectrum of available data because of the dense social net of relations between community members. Important CIKS data for modeling Groups are explicit self information and implicit tree-like communication structures. In addition to that, highly dynamic contextual data such as locations and velocities offer the possibility to model Ad-Hoc Groups. Such data can be accessed with present and future technologies such as satellite based systems and cell based location. Privacy and pragmatics of using location data need to be ensured through appropriate policies. Since location and velocity data are not available in the required precision and with the required meta-knowledge, we need to simulate the data collection with the help of stochastic models. Thus we developed the SUMI mobility model which allows for simultaneous simulation of group motion and individual motion. The special feature of SUMI is that the underlying abstract group structures and instantiating Ad-Hoc-group structures are known with all their parameters at any time step of the simulation. This is due to an appropriate pre-simulation schedule generation process. Because of the thorough design of the involved stochastic processes, the simulation is very detailed and realistic and is much better suited for individual mobility simulation in urban areas than any other simulation available. Finally, basic considerations show that velocity and location should be treated on equal footing and modelling and distance measures should reflect that appropriately.

Chapter 4

Non-Contextual Data: Explicit Self Information and Communication Data with Tree-like Structure

As an example for explicit self information, this chapter discusses lists and sets of interest utterances of users. After a general introduction, we discuss general characteristics of such interest phrases with the help of four specially collected test collections of interest phrases. We will then discuss two major forms of such explicit self information: lists-of-choices interest vectors and sets of free text interest phrases. For each of these subtypes, similarity measures are proposed and extensively discussed. The similarity measures for lists-of-choices vectors are based on given taxonomies. The measures for free text phrases are based on semantic nets and statistical natural language processing. The final part of the chapter deals with textual communication data with a threaded, tree-like structure which can be typically found in communities. We will develop a similarity measure that allows to compare persons on the basis of their communication behaviour. The measure uses posting content analysis as well as algebraic properties of the tree-structure.

4.1 Interest Phrases

As has been motivated at the beginning of the previous chapter, the next data source we will investigate are lists (vectors, sets) of interest phrases or -keywords. These lists are explicitly gathered e.g. with the help of HTML forms (text-fields or choices from a list) mostly in the course of the build up of the personal profile after having registered at an online community platform. As a key part of the personal profile they usually have the function of directly informing other users about one's interests. Another way to use these lists of interest-phrases or -keywords is for matchmaking [87] as in case of dating communities like Friendscout24.de [127]. They can also be used as proactive information retrieval queries etc..
What is interesting about such explicit lists of elements characterizing a user is that on the one hand they represent valuable self-information and on the other hand, every element (text-phrase or single word) of such a list usually corresponds to a single concept and is an answer to a defined question (e.g. "What are your interests?" or a semantically similar question). This property is especially interesting when comparing two of these lists because the heuristics of comparison can be fine tuned with respect to the question. Even more important is the property of such

lists that their elements (aspects, concepts, answers) are clearly separated. This is not the case with monolithic blocks of free textual answers which may contain several aspects, concepts or answer elements in intertwined natural language structures, using the full expressiveness of natural language (negations etc.). Extraction of these answer elements out of compact blocks of text requires complicated natural language analysis.
Another aspect that makes lists of interests especially interesting is that they can logically not contain negations because the implicit question that generates such lists will usually not allow for negated concepts as answer elements. Although it is possible to additionally ask e.g. "what do you not like?", it will generally not be the case that positive and negative answers will be mixed in one list, which generally allows to identify an interest phrase with a positive concept C instead of having to investigate $\neg C = \mathcal{U} - C$ that is the complement of C with respect to a universe $\mathcal{U}$ of answer alternatives. Negations can generally be assumed to add a substantial level of difficulty to declarative logical reasoning and (statistical) heuristics as well.
The basic distinction we will make is between lists of choices and lists of free text answer elements. In this section, we will give examples for both variants and discuss access, test-corpora, structure and comparisons and similarities of such lists taking interests as an example.

4.2 Accessing Interest Phrases

Usually, communities use a single platform which allows for central access to the profile structures of all users. Distributed architectures using agent paradigms must include means to exchange these information in a standardized format (e.g. XML) possibly supplemented by a privacy negotiation mechanism (see e.g. [194]).
In case of free text input of interest phrases or interest keywords, HTML forms are usually used. In order to achieve a clear separation between the elements of the answer vector, it should either be made explicit that the elements should be separated with a special separation character (like ';' or '||') or separate input fields for the list elements should be used. It is always possible to allow for an option to enlarge the number of such input fields, so that the separation of list elements does not represent a limitation with respect to the number of such elements.
We will now investigate four examples of interest vectors or sets.

4.3 Lists Of Choices

If users must chose among a given list of answer alternatives, in our case a given list of interests we will call the resulting answer list a List of Choices or List-of-Choice Vector.
As an example, 100 interest vectors have been downloaded from the dating community "yahoo.personals" [134]. With the help of the search function, a list of women in the New York area aged 23-49 was retrieved and the interests of the first 100 women have been downloaded from their personal profile. After registration at the dating community, members are given the possibility to construct a personal profile. In the course of this build up process, a user can chose as many interests from a given list as desired. Figure 4.1 shows the list of alternatives and a part of the 100 downloaded vectors.
We will refer to this collection in the rest of the chapter as "Dating Collection".

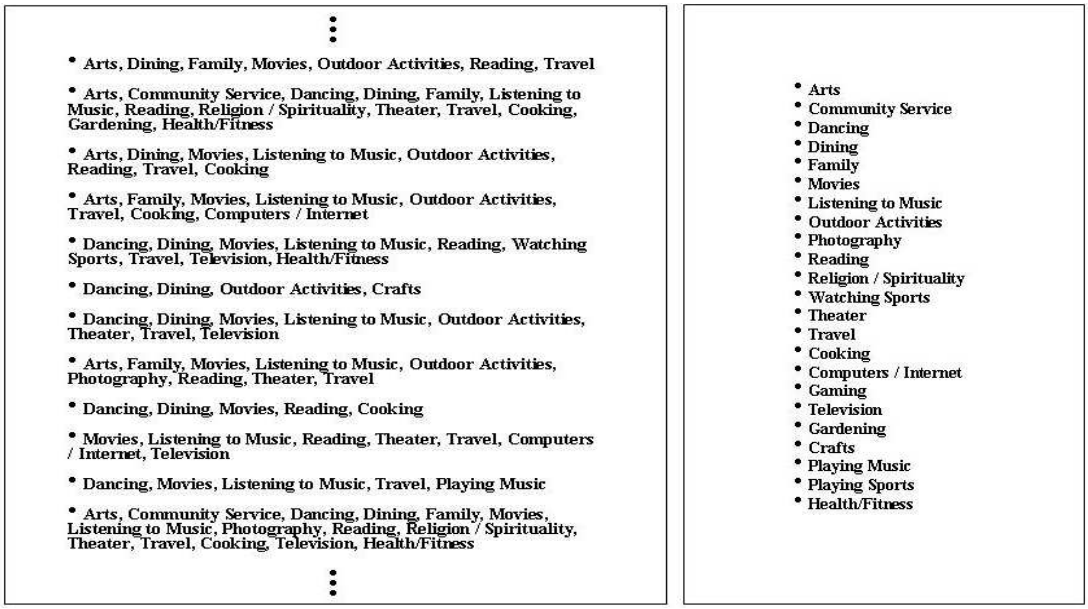


Figure 4.1: Example for a List Of Choices. Left subfigure: A subset from the 100 downloaded list-of-choice interest vectors ("Dating Collection"). Right subfigure: List of alternatives to chose from.

4.4 Lists of Free Text Elements

If free text entries are possible for the elements of the list, additional difficulties arise that will be discussed in section 4.5. We have therefore collected 3 sets of such free text interest sets to be able to compare different varieties of such lists and to identify common problems.

4.4.1 MTV Collection

From the community of the Music Television station MTV [135] we have collected interest sets from 100 users who posted contributions in the discussion board "Fight for your rights" which was randomly chosen from the list of available discussion boards. The users of this community are typically between 17 and 25 years of age and have a strong affinity to music and youth culture in general. The community's CIKS consists mainly of discussion boards and user profiles augmented with an extensive editorial content. Several elementary services are provided like searching the user database, chat etc..
After having registered at the community, a user can enter his interests which are simply called "keywords" in a large free text area. No explicit hints concerning the delimiter / separation sign to use are given. The question is simply stated as "List a few things you're really into (Examples: snowboarding, Yankees, Incubus)!". Although most users use commas to separate their interests, some users use other delimiters such as '.', '-', or "..." or no delimiters at all. Since list element separation is a key prerequisite of the present considerations and since a separation can always be achieved during the input phase (as has been justified above), the user input was manually preprocessed by replacing all delimiters by ';' to simulate the results of an input method which emphasizes separation. Due to an error in the HTML display of the interests, some words were incorrectly cut in half by spaces. Since this does not represent an error on the

user side, we manually corrected these errors.
We will now look at some examples and analyze some difficulties in connection with the input of these answers.

1. *VOLLYBALL, SHO PPING, PHONE, HANGING OUT, CHEERING, SURFING, SNO WBORDING, SKII NG*
   A typical list of interests mostly in form of single words. Words cut in half due to HTML formatting errors were corrected in an obvious way. "hanging out" is an example for a simple phrase of two words. Spelling errors like "SNOWBORDING" were not corrected at the manual preprocessing stage, because they represent user created errors which will naturally occur.

2. *humanist, pro-monogamy, anti-racist, anti-sexism, omnivore, anti-liberal, anti-conservative, promoter of "bad words", non-patriot, refuser of all religions, questioner, philosopher, and i think agreeing with society is often a bad sign*
   This interest set is rather a self description than a list of interests. This can be avoided by a more clear question. The last phrase is a complete sentence. This form of an "interest phrase" will also probably not occur when the question is posed more clearly and the form of input fields is chosen more appropriately. Commas have been manually replaced by ';'s.

3. *I love to snowboard, this will be my 5th year at it. I am trying to land a 540 in the pipe but just about broke my arm last year trying. I also like paintballing, soccer, long romatic walks, foreplay, and of couse the ladies!*
   A similar problem occurs with this list. The single free text input field was "misused" to create a prose text. Even in the last part which is more like a list of separated concepts filling phrases such as "of course" occur which would probably be omitted if the desired list character of the input would have been made clearer. The '.'s have been replaced by ';'s.

4. *1. Netball 2. Basketball 3. Hangin out*
   Again a problem with alterative delimiter signs, here in form of an enumeration. Changed to ';'.

5. *big cats Baby Rasta moving graduating*
   No delimiters at all. Insertion of ';'.

6. *uhm hanging with my friends and partyin and just havin fun*
   Here, we have a complete sentence instead of a list of phrases. The "and"s have been replaced by ';'s. Complete texts tend to contain onomatopoetic particles like "uhm" which were probably included to give the list a more personal "spoken" touch. Can be avoided if the input possibilities are discretised as suggested above.

7. *PUNK ROCK PUNK ROCK*
   The repetition of important concepts is probably intended as a means of amplification. The user would probably also have stated this interest multiple times if discrete input fields or a clear hint for a distinct separation character would have been provided, so the two instances of "PUNK ROCK" were separated with a ';'.

8. *Harley Choppers, Dark Poetry, Horror films, walking in cemeteries under a full moon, haunted houses, woodworking (making coffins), reading all types of books especially E.A.*

Poe and Lovecraft. Gothic Music especially Ethereal Gothic
A list that contains more complicated phrases. Delimiters were replaced by ';'s.

9. *Good Charlotte, 12 Stones, Die Trying, Mest, Linkin Park, Disturbed, Rancid, New Found Glory, Simple Plan, Chevelle, AFI, Brand New, Story of the Year, All-American Rejects, Fall Out Boy, Brand New, The Used, 3 Doors Down, Sinedown, and many more*
Band names are very typical as interests in this community like this set clearly shows. Formulations like "and many more" are less probable in case of a discrete input-form.

4.4.2 Party Community Collection

In order to verify the findings of the MTV collection a similar collection from an open "Party People Community" named "MyScene" [136] was gathered. The community is an open forum mostly visited by "Ravers" (young people interested in electronic music). The profile generation HTML form provides a text field for inputting the interests. The text field is labeled "interests" and is only one line high. From the list of registered users, we collected 100 interest sets.
Although basically the same phenomena and problems occur than in case of the MTV collection, one can see a slight tendency to be more regular in terms of delimiters and the tendency to "misuse" this profile parameter in form of prose text self descriptions is slightly smaller. This may be due to the input field which is only one line high and thus does not mislead the users towards writing prose texts.

4.4.3 Survey Collection

In order to further investigate the gathering process of free text answer sets, a collection of another 100 interest sets was collected with the help of an online survey. The online survey's text is shown in a screenshot the left upper subfigure of figure 4.2. This text was sent to all user's in the email-tool's address book of the thesis author, of three student research assistants working in the COSMOS project (R. Friess, C. Ehmig, M. Geiger) and of a PhD-student (W. Gersten) at Daimler Chrysler Research, Ulm. The evaluation of the survey was stopped after 100 sets had been collected. The survey used the open source tool Open Survey Pilot [138].
Although in the narrow sense the participants of the survey do not represent a community there are several common characteristics: Most of the participants are affiliated to universities and scientific research. Most of the participants are young people and most of the participants are socially connected in real life.
The resulting sets showed that people were very disciplined in using a single separation character, although a large text-field was provided. This might have been due to the artificial character of the survey (see text in figure 4.2), but might as well be attributed to the clear formulation of the question with the hint to use this separation character in a disciplined way. In concrete realizations of the suggestions of this thesis, the users can be expected to adapt their input behavior with respect to separation if the services that rely on this separation (e.g. Ad-Hoc-Group based services) offer an added value for them (normalizing effect).

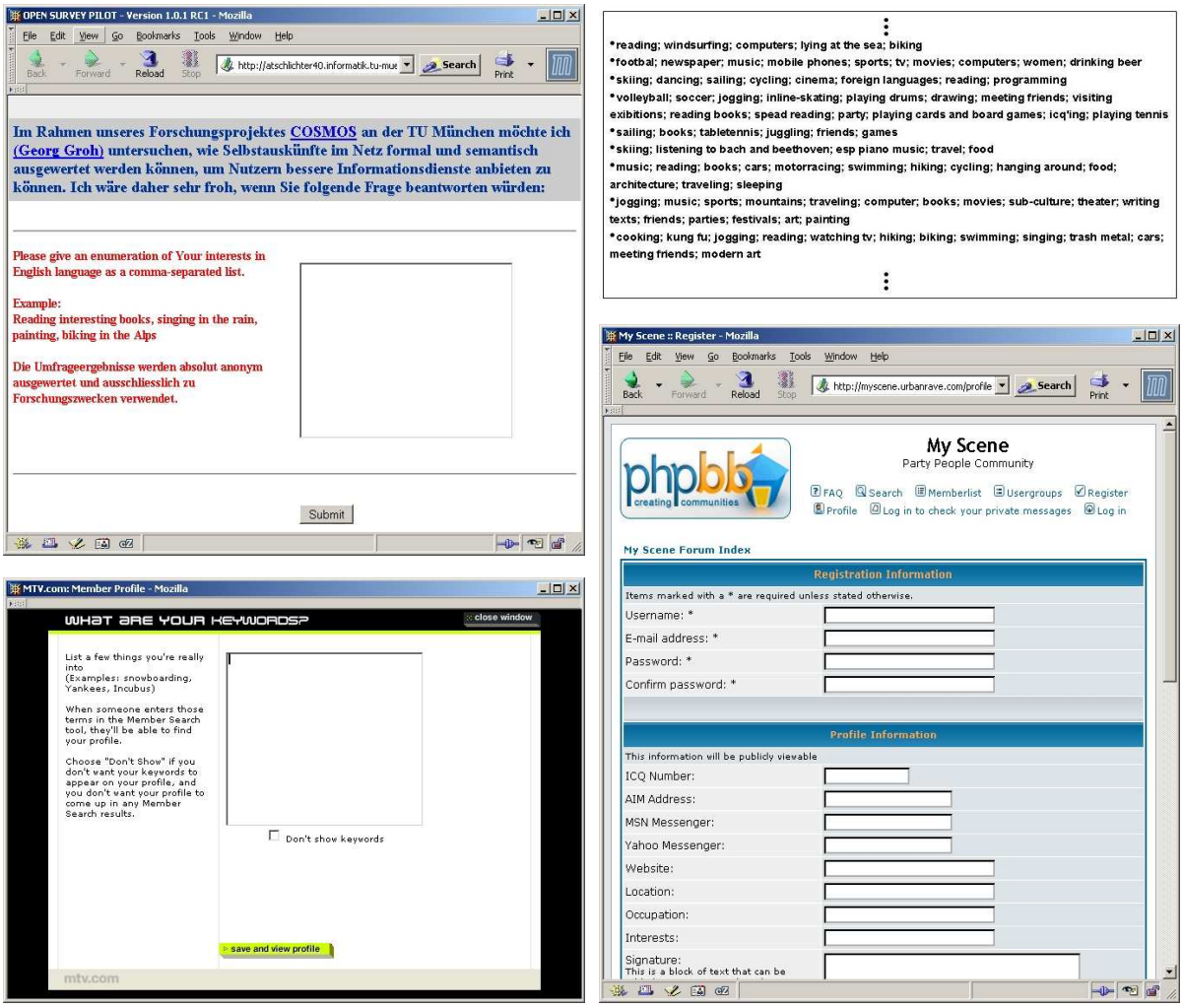


Figure 4.2: Left upper subfigure: Screenshot from the survey for collecting interest sets. Right upper subfigure: A subset of the sets which have been collected in the survey. Left lower subfigure: Editing of interests "keywords" in the MTV community. Right subfigure: Editing of interests (and lots of other personal profile information) in the MyScene community.

4.5 Similarity Measures

Information that has been directly and explicitly provided by a user should be regarded as the most valuable form of information about that user (in contrast to implicitly deduced information (which always involves heuristics and other algorithmic characteristics)) and should therefore be evaluated as thoroughly as possible.
In case of analyzing sets of interest keywords or -phrases with the goal of finding clusters of users with similar interests, this means that besides a purely syntactical analysis, the semantics of the provided keywords should also be considered as far as possible.
We will now investigate how such vectors or sets of interests can be compared with respect to group detection. In order to compute a distance or similarity measure that can be used in a clustering algorithm we will have to combine statistical measures and semantic measures. First,

a similarity measure for lists of choices will be investigated. We will then investigate the case of free text interest phrases.

4.6 Similarity Measures for Lists of Choices

If users are given answer alternatives for a question, these answer alternatives form a simple **ontology**, were every answer alternative corresponds to a concept. See Appendix A for a brief introduction into the concept of - and definition of the term ontology.

4.6.1 Ontologies for Lists of Choices

Ontology languages or -formalisms often lack a precise semantics. Assuming a rather liberal point of view towards what is regarded as a formalism or language for the definition of an ontology, these formalisms or languages include database oriented frameworks such as the calculus of Relational Algebra and Entity Relationship diagrams [83] as well as frameworks from software engineering and object oriented design such as UML [15] as well as frameworks originating from Artificial Intelligence such as Frames or Semantic Nets [159, 58] as well as frameworks from the field of Semantic Web such as RDF(S) [66], DAML+OIL [29] or OWL [116]. A precise semantics for these formalisms or languages is generally given in form of a suitable fragment of first oder logic (FOL) such as a Description Logic (like $\mathcal{FL}^-$) [42] or directly in FOL which is an approach we will also use in the following discussion. Figure 4.3 shows two alternatives for **simple subject taxonomies** or ontologies for the interest choices of figure 4.1.
If we regard each entry in the list-of-choice alternatives as a **concept or class** of the ontology, we assume that these concepts represent sets of concrete interests which can be classfied as belonging to this class or concept. A class is an **unary predicate** C which under an **interpretation** I is assigned a set $C^I = \{x|C(x)\}$ of objects of the universe of discourse. As an example consider the concept "reading". This concept includes "reading books from Heinrich Böll", "reading the Süddeutsche Zeitung" etc. which, in turn, can already be regarded as instances (objects) or as concepts themselves which would make "reading 'Billard um Halbzehn"' an instance of "reading books from Heinrich Böll".
The concepts from the list can be generalized in form of **super-concepts or super-classes** as it is shown in figure 4.3. As projects in the domain of common sense ontologies show, there are generally many possibilities to construct a hierarchy of super-concepts for a given flat list of such classes, especially if we use only a definite unambiguous is-a abstraction relation like it is the case in figure 4.3. The figure shows two possibilities to construct super-classes.
The **"is-a" relation** (semantically: Abstraction; lexically: Hyponymy [1]), which is depicted by arrows in figure 4.3 is defined by logical implication: $Gardening \rightarrow ManualActivity$ (which in Description Logics corresponds to subsumption: $Gardening \sqsubseteq ManualActivity$). Implication is interpreted by a subclass relation: $Gardening \rightarrow ManualActivity$ is true iff $Gardening^I \subseteq ManualActivity^I$.

As it is well known from the formalisms mentioned at the beginning of this section, there are several other types of associations that can possibly occur in ontologies. Prominent examples are **"instance-of", "part-of" and "property-of" relations**. "Part-of" (semanti-

[1] Relation is read from left to right: "cat" is a Hyponym of "mammal": "cat" $\xrightarrow{\text{"is-a"}}$ "mammal". The inverse relation is Hypernymy: "mammal" is a Hypernym of "cat": "mammal" $\xrightarrow{\text{"super-class-of"}}$ "cat"

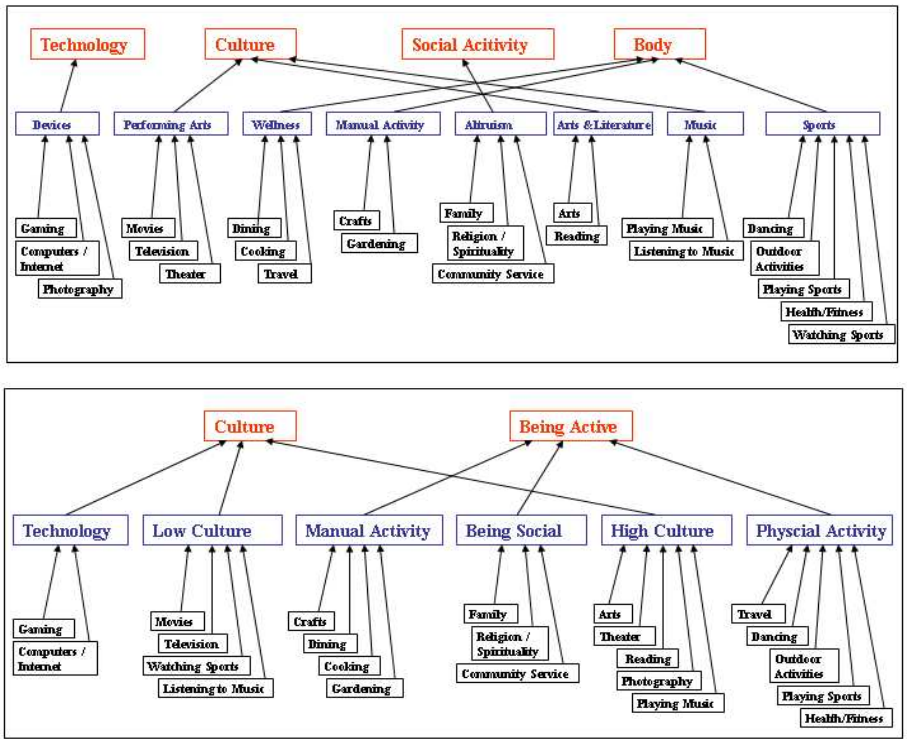


Figure 4.3: Two possible simple subject taxonomies for the interest choices shown in the right part of figure 4.1. Arrows indicate "is-a" relations between the concepts.

cally: Aggregation, lexically: Meronymy [2] and "property-of" relations are defined by binary predicates: $Rim \xrightarrow{\text{part-of}} Wheel$ is defined by $\forall x(Wheel(x) \rightarrow \exists y(PartOf(y,x) \wedge Rim(y)))$ and $Diameter \xrightarrow{\text{property-of}} Wheel$ is defined by $\forall x(Wheel(x) \rightarrow \exists y(PropertyOf(y,x) \wedge Diameter(y)))$. For a more precise definition, we need to include further theories on part-whole-systems in order to define the binary predicate "part-of" more precisely.
Instance-of relations like e.g. $madonna \xrightarrow{\text{instance-of}} Pop$ intend that the concept (here Pop) is interpreted as a set which contains a certain individual. They are defined as $\exists m(Pop(m) \wedge \exists madonna(Name(madonna,m) \wedge String(madonna)))$ which restricts fulfilling interpretations in the desired way (see [42] for more examples).
Other types of semantic relations such as "functionally-related-to" (like e.g. in penguin - Antarctica) may be liberally subsumed under "part-of" relations if the one domain of discourse allows for such as point of view (e.g. in marine biology a penguin is certainly not "part-of" the ocean, whereas in an ontology defining characteristics of the world's regions it might be o.k. to take this point of view).
Figure 4.4 shows two more (interest-)ontologies which additionally make use of "part-of" and "instance-of" relations: A music taxonomy and a taxonomy of car parts. Interesting edges are marked with numbers $(1), (2), \ldots$. In the music taxonomy, the elements at the lowest level may be regarded as instances but may as well be regarded as concepts (classes). If the former is

[2] Relation is read from left to right: "motor" is a Meronym of "car": "motor" $\xrightarrow{\text{"part-of"}}$ "car". The inverse relation is Holonymy: "car" is a Holonym of "motor": "car" $\xrightarrow{\text{"whole-of"}}$ "motor"

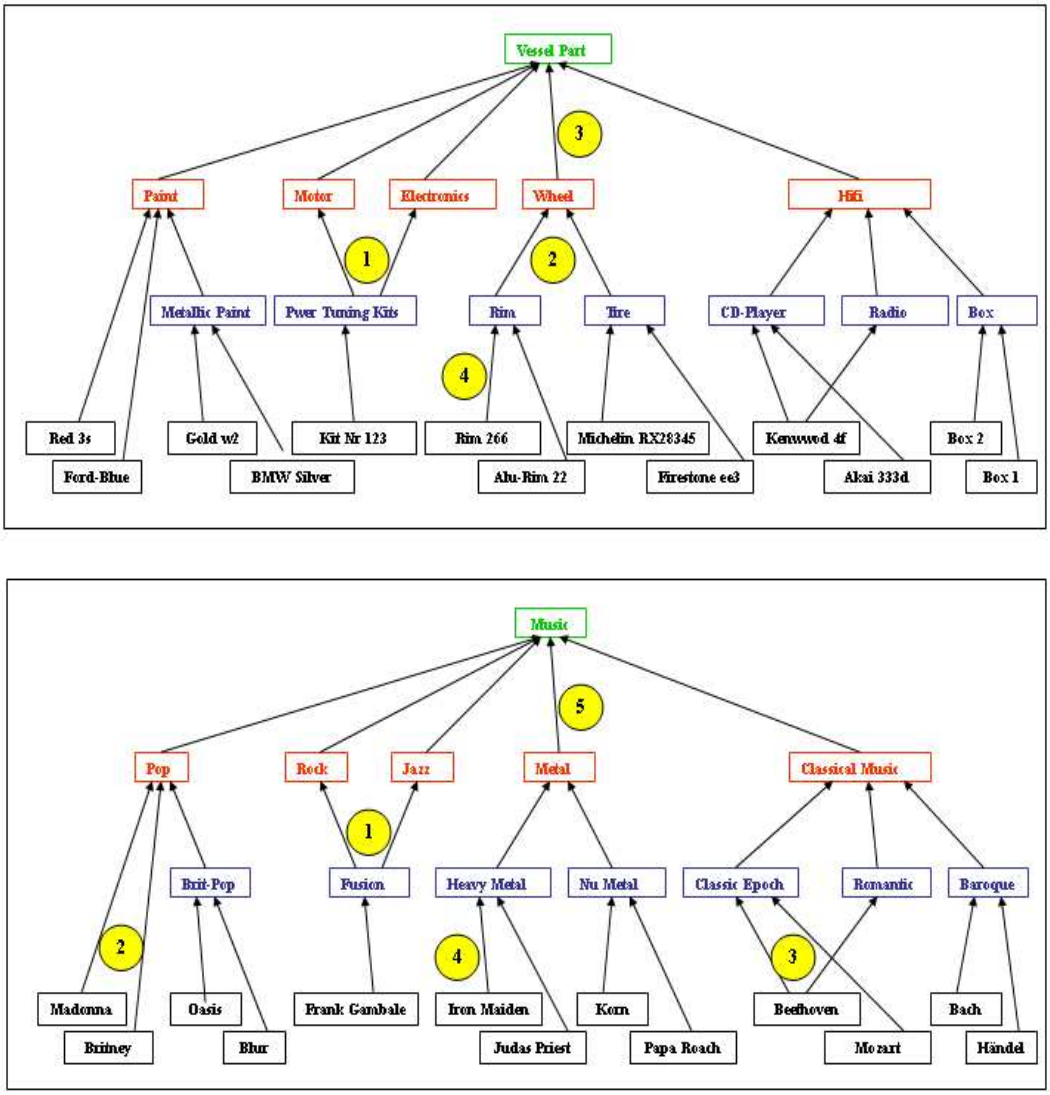


Figure 4.4: Two further interest area ontologies that make use of "is-a" relations, "part-of" relations and "instance-of" relations. Interesting relations are marked with numbered circles and are referred to in the text.

true then relations (2) are "instance-of" relations otherwise "is-a" relations or even "part-of" relations. If the upper category of (2) would be "pop-singers" then "part-of" would not be adequate. Relation (5) shows a conventional abstraction ("is-a"). While (2) is an "instance-of" / "is-a" relation with a concept at a higher level of the taxonomy, relation (4) connects to a lower level. Relation (1) shows an example of a multiple "instance-of" / multiple inheritance ("is-a") relation. Roughly the same considerations apply to the car-part taxonomy where the lowest level is clearly a level of instances which makes (4) an "instance-of" while (1), (2) and (3) are clearly "is-a" relations.

In a particular community, the common pursuit which is identified with the pursuit to build a CIKS with a special focus (see chapter 1) allows to construct a **specialized simple taxonomic ontology for certain profile elements** such as interests. In case of the dating community,

the ontology does not seem to be very specialized. But keeping in mind that the CIKS consists mostly of profiles intended to give a possible partner a quick overview over a person, the ontology and its level of specification seem to be well adapted to the common pursuit of the community. The more specific the interests of the community are, the more specific the interest ontology will be. Except for interest this generally applies to all elements of the CIKS where such an ontology is basis for choosing answer elements. As has been pointed out in section 1.3.3 a community can act as an information or knowledge layer between the individual user and the web because the CIKS is a manifestation of a special common pursuit or interest. With that in mind, it becomes a feasible task for the community members (or its administrators) to **limit the domain** for which ontologies have to be created. The community chooses the taxonomic structure (e.g. one of the alternatives of figure 4.3) according to its preferences and informational needs.
In case of interests we can assume that the special type of relation connecting the elements of level n of the taxonomy with the elements of level $n-1$ is one of "is-a", "instance-of", and "part-of". For the intended application, it is of minor importance to which type of relation the edges of the graph correspond to. Either of the three represents a form of generalized abstraction. We will therefore call a relation which is of type of one of these 3 relations a **generalized abstraction relation**.

4.6.2 List-of-Choice Similarity Measure in Topic Taxonomies with Generalized Abstraction Relations

We will assume that the set of choices for vectors of interests can be identified with a simple ontology (a topic taxonomy) which uses generalized abstraction relations as a construction means. How can vectors of interests that are made up of elements of such a taxonomy be compared with one another with high expressivity?
Let the **graph of the taxonomy** be a "tree like" (directed, connected, acyclic) graph $G = (\mathcal{E}, \mathcal{V})$ of height m. (Usually trees are defined as connected and acyclic undirected graphs where due to acyclicity "multiple inheritance" is not allowed. Multiple inheritance (a node has more than one outgoing edge) is allowed in directed, connected and acyclic graphs which we call "tree-like".) The Graph has concepts $C \in \mathcal{V}$ as vertices of level $0 < n < m-1$, concepts and instances (objects) as leafs (level $n = m-1$), and generalized abstraction relations $R \in \mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ as edges. A **list-of-choices vector** x will be a vector $x \in \{0,1\}^{|\mathcal{V}|}$ with

$$x_i = \begin{cases} 1 & \text{if } C_i \text{ has been chosen} \\ 0 & \text{if } C_i \text{ has not been chosen} \end{cases} \tag{4.1}$$

As an example consider a flat tree with one abstract root C_0 and vertices C_1 ="cooking", C_2 ="reading", C_3 ="arts" and C_4 ="dining". If the choices are "arts" and "cooking" the list-of-choices vector will be $x = (0,1,0,1,0)$ where the first 0 corresponds to the abstract root and is of low significance because it is assumed to correspond to the overall area of interest of the community. If it is offered as a choice option, it should be included, if not, it can be left out. The easiest way to compute a **similarity measure** $\mathrm{sim}(x^{(j_1)}, x^{(j_2)})$ **between lists of choices vectors** is to ignore the taxonomic structure (its edges $\mathcal{E}$) completely and compute a simple normalized inner product:

$$\mathrm{sim}(x^{(j_1)}, x^{(j_2)}) = \frac{1}{||x^{(j_1)}||\ ||x^{(j_2)}||} \sum_{i=0}^{|\mathcal{V}|-1} x_i^{(j_1)}\ x_i^{(j_2)} \tag{4.2}$$

By normalizing the list-of-choice vectors to unity in (4.2), we make vectors with few elements qualitatively comparable to vectors with many elements. Under this paradigm, the total interest "intensity" is assumed to be equal for all persons. This means that a person stating e.g. only one interest choice puts the whole interest "intensity" in this particular choice whereas somebody that states e.g. 10 interest choices will only be able to devote one tenth of his unit interest "intensity" to each of the ten choices. This paradigm neglects the fact that the overall interest "intensity" can be different for different persons. One the one hand, people may, in general, not all have the same overall personal quantum of interest "intensity". This quantum may depend on individual factors like personality, character or intelligence. On the other hand, one person might have a greater interest overlap with the community's common pursuit than another person while both have the same overall interest "intensity". Nevertheless, these two effects can be assumed to be small enough to justify normalization to unity. Otherwise, people would have to state their overall level of interest "intensity" explictly, which is not desirable.

Since in (4.2) we completely ignored the set of vectors from other users and the edges (the taxonomic structure), the similarity measure has the potential to be improved.
The **first improvement** is an adaptation of the Inverse Document frequency ***idf* measure** of information retrieval's vector model [6]. The vector model represents a text-document by a vector of weights where each weight corresponds to a word of the text's vocabulary. Words that occur more frequently (that have a higher so called term frequency tf) have a higher weight than words with a lower term frequency. The *idf* measure of a word with index i is computed by considering all documents in the collection. The expression for *idf* is

$$idf_i = \log(N_i/N) \tag{4.3}$$

where N_i is the number of documents that contain the word with index i and N is the total number of documents in the collection. The *idf* measure is multiplied with the term frequencies to yield the overall weights of a vocabulary element and is a measure of the information content of the word with index i. If the word occurs in a large fraction of the documents, its *idf* measure will be low because its discriminative power corresponding to the benefit of knowing that the word occurs in a particular document (its **information** (see [51])) is low.
If we transfer the idea to our case, we can switch from binary "weights" to *idf*-weights for the list-of-choice vectors. If we have N binary list-of-choice vectors $x^{(0),\mathrm{bin}}, x^{(1),\mathrm{bin}}, \ldots, x^{(N-1),\mathrm{bin}}$ we have then:

$$x_i^{(j),idf} = \begin{cases} \log \frac{N_i}{N} & \text{if} \quad x_i^{(j),\mathrm{bin}} = 1 \\ 0 & \text{if} \quad x_i^{(j),\mathrm{bin}} = 0 \end{cases} \quad \text{with} \quad N_i = \sum_{k=0}^{N-1} x_i^{(k),\mathrm{bin}} \quad . \tag{4.4}$$

The **second improvement** takes into account the edges of the graph, that is the **taxonomic structure**. Regarding the choices in figure 4.1 and e.g. their taxonomic generalization in the lower subfigure of 4.3 we see that in case of the dating community, the list-of-choice interest vectors only make use of the leaf concepts. If one vector was ("arts", "gardening", "playing music") and the other vector was ("crafts" "listening to music", "community service") the similarity computed with equation (4.2) is zero although "gardening" and "crafts" and "playing music" and "listening to music" are "similar" on an upper level of the tree-like graph because both pairs have a generalized abstraction relation with an identical upper class. Even if the users can make their choices from the whole taxonomy and not just the leaves, the "similarities" of choices on upper levels should be regarded.

The idea for a solution is simple: for each initial list-of-coice interest vector (binary or idf weights), compute separate partial lists of choice interest vectors on each level of the taxonomy graph by "propagating" choices upward. Normalize each of these level-vectors and compute a weighted sum of inner products of corresponding level-vectors of the both list-of-choice interest vectors that are to be compared.
We will first give a precise algorithmic definition of this procedure and then present an example. For notational convenience, we define a function f that maps the plain index $i \in [0, |\mathcal{V}| - 1]$ of a concept in the tree-like graph to the level of the graph $l \in [0, m-1]$ that this concept is in. The function follows trivially from the taxonomic structure of the graph.
The algorithm's input is two list-of-choice vectors $x^{(j_1)}$ and $x^{(j_2)}$ (either binary or *idf* weights). For each of the two vectors, the algorithm's intermediary output is a set of m list-of-choice level-vectors $\{x^{(j),(0)}, x^{(j),(1)}, \ldots, x^{(j),(m-1)}\}$ (one for each level of the graph) which represent the choices constrained to each level, supplemented by weighted contributions from the lower levels. The similarity measure is then computed on these two sets of list-of-choice level-vectors as a weighted sum of level-wise products.

The formal algorithm is shown figure 4.5 is in fact very simple but just requires some inconvenient notation in its precise formulation. Figure 4.6 is intended to clarify the computation of list-of-choice level-vectors in algorithm by giving an example. The left part of the figure shows the taxonomy graph with the indices of the concepts C and the levels. The nodes which have been chosen in the example list-of-choice vector $x^{(j_1),\text{bin}} = (0, 1, 0, 0, 1, 1, 1, 0)$ are shown in a darker color. For clarity reasons, binary weights have been chosen for the example vector.
The right side of the figure shows an explicit representation of the graph, the level function and the level weights. Especially important are the three list-of-choice level-vectors $x^{(j_1),(l)}$ which are constructed from $x^{(j_1),\text{bin}}$. Each of the level-vectors contains only entries for vertices on its level. The weights of each of these concepts (vertices) is computed by summing up the weights from nodes in the next lower level which are sub-concepts of this concept. A decreasing weighting of the contributions from the lower level to compensate for the increasing size of the numbers toward the root is not necessary, because the similarity measure uses normalized level vectors. Thus only the directions of the level-vectors matter.
In the computation of the similarity measure (which is not shown in figure 4.6) the similarity contributions from the pairs of level-vectors are weighted with the weights γ_l. These weights exponentially decrease from the leafs towards the root. The philosophy behind that is that interest concepts at higher abstraction levels do have the same expressivity with respect to computing similarity between persons as interest concepts on lower abstraction levels. It is more expressive to find that two persons are both interested in "Madonna" than to discover that both are interested in "Music".
In chapter 5 we will investigate how this similarity measure performs on the dating-collection.
We will now investigate how similarity measures for free text interest phrases can be constructed.

4.7 Similarity Measures for Free Text Interest Phrases

If users are allowed to use free text phrases to describe their interests, the problem of comparing the resulting lists of free text phrases is an order of magnitude more complicated than in case of lists of choices. E.g. if users choose from given alternatives, the problem of correct **spelling** is not relevant which is not the case with free text phrases. Furthermore, we have the possibility of **concept separation** errors as was discussed in section 4.4, where we have made plausible that

$\mathrm{sim}(x^{(j_1)}, x^{(j_2)})$

- **Input:**
  - **Directed, connected, acyclic Graph:** $G = (\mathcal{E}, \mathcal{V})$ **of height** m;
  - **Two list-of-choice vectors** $x^{(j_1)}, x^{(j_2)}$ **(binary or idf):** $x^{(j_1)} = (x_0^{(j_1)}, x_1^{(j_1)}, \ldots x_{|\mathcal{V}|-1}^{(j_1)})$, $x^{(j_2)} = (x_0^{(j_2)}, x_1^{(j_2)}, \ldots x_{|\mathcal{V}|-1}^{(j_2)})$;
  - **Level function:** $f : [0, |\mathcal{V}| - 1] \to [0, m-1]$.
- **Set weights** γ_l **for each level of the graph:** $\gamma_l := 1 \,/\, 2^{m-l-1}$.
- **Initialize: Set**

$$x_i^{(j_{1,2}),(m-1)} := \begin{cases} x_i^{(j_{1,2})} & \text{if} \quad f(i) = m-1 \\ 0 & \text{if} \quad f(i) \neq m-1 \end{cases}$$

- **Initialize: Set**

$$\mathrm{sim} := \frac{1}{||x^{(j_1),(m-1)}||\ ||x^{(j_2)(m-1)}||} \sum_{i=0}^{|\mathcal{V}|-1} x_i^{(j_1),(m-1)}\, x_i^{(j_2),(m-1)}.$$

- **For** $(l := m-2;\ l \geq 0;\ l$**--) //iterate over graph levels**
  - **For** $(i := 0;\ i < |\mathcal{V}|;\ i$**++) //iterate over vector index**
    - **Set**

$$x_i^{(j_{1,2}),(l)} := \begin{cases} x_i^{(j_{1,2}),(l)} \ + \ \sum_{\{k|(C_k,C_i)\in\mathcal{E}\}} x_k^{(j_{1,2}),(l+1)} & \text{if} \quad f(i) = l \\ 0 & \text{if} \quad f(i) \neq l \end{cases}$$

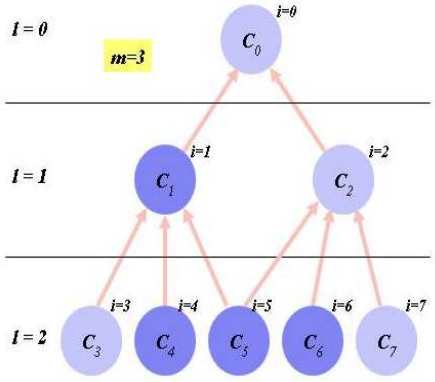
    Endfor
  - **Set**

$$\mathrm{sim} := \frac{1}{1+\gamma_l} \left(\mathrm{sim} \ + \ \gamma_l \ \frac{1}{||x^{(j_1),(l)}||\ ||x^{(j_2)(l)}||} \sum_{i=0}^{|\mathcal{V}|-1} x_i^{(j_1),(l)}\, x_i^{(j_2),(l)} \right).$$

  Endfor
- **Return** sim.

Figure 4.5: Algorithm for a Similarity Measure $\mathrm{sim}(x^{(j_1)}, x^{(j_2)})$ between List-of-Choice Vectors in a Topic Taxonomy with Generalized Abstraction Relations.

the problem can be overcome by means of a clear user-interface design. Apart from that, in case of lists of choices the taxonomy of possible choices is a given quantity because it is determined by the community. The community decides about the meaning of the taxonomic relations and



- Graph $G = \big(\mathcal{V} = \{C_0, C_1, C_2, C_3, C_4, C_5, C_6, C_7\}, \mathcal{E} = \{(C_1, C_0), (C_2, C_0), (C_3, C_1), (C_4, C_1), (C_5, C_1), (C_6, C_2), (C_7, C_2)\}\big)$ of height $m = 3$.
- Level function: $f(0) = 0,\ f(1) = 1,\ f(2) = 1,\ f(3) = 2,\ f(4) = 2,\ f(5) = 2,\ f(6) = 2,\ f(7) = 2.$
- Level weights: $\gamma_2 = 1,\ \gamma_1 = 0.5,\ \gamma_0 = 0.25.$
- $x^{(j_1),\text{bin}} = (0, 1, 0, 0, 1, 1, 1, 0).$
- $x^{(j_1),(2)} = (0, 0, 0, 0, 1, 1, 1, 0),$
  $x^{(j_1),(1)} = (0, 3, 2, 0, 0, 0, 0, 0),$
  $x^{(j_1),(0)} = (5, 0, 0, 0, 0, 0, 0, 0).$

Figure 4.6: An example for the first part of the algorithm of figure 4.5

the similarity measure can be fine-tuned with respect to this meaning. In the previous section we gave an example for such a similarity measure for taxonomies with generalized abstraction relations. For free text phrases, the **ontology is implicitly determined by the users** and we cannot assume that the relations between its concepts are known a priori as in case of a fixed ontology. Using the same approach as in case of lists of choices we would have to design the similarity measure in a way that would be flexible enough to handle all possible types of relations. In addition to that, in case of comparing lists of free text phrases, it is not clear which concept of one vector is to be compared with which concept of the other.
We will now shortly discuss these problems and present a pragmatic solution for each problem with respect to our problem domain.

4.7.1 Spelling

Spelling errors in the lists can be automatically corrected by means of **sophisticated spelling correction techniques** which are very briefly reviewed in appendix B. It can be assumed that all techniques which can be accomplished with modest resources (that is non-word error detection and isolated word spelling correction (see appendix B)) will deliver only modest accuracies. Diverse isolated word spelling correction algorithms deliver an accuracy between 60 and 80 % [94]. From various surveys on spelling correction one can conclude that the expenses that go along with attempts to improve the performance of automated spelling correction (e.g. by context-sensitive methods) will not scale linearly with the gain in accuracy. For a near perfect spelling correction, the algorithm would have to be as complex (order of magnitude) as an algorithm for near perfect semantic understanding and representation of a text, which is still an unsolved or even unsolvable problem.

When investigating typical spelling error rates one has to carefully determine the **conditions** under which these error rates occur. These conditions include the level of typing experience of the text creators, their educational level, the purpose of the text-input, the technical environment used for inputting the text and many more factors. In [94], a diversity of investigations is reviewed, reporting from under 1 percent error rates in published texts like examples from AP

newswire to reporting around 6 percent error rates in typed textual conversation (chat).
In case of sets of interest phrases we can roughly **estimate the error rates** by considering our free text interest sets collections. Unfortunately, the determination of what is actually an error is more complicated than in case of the experiments mentioned above. When determining if a certain word is spelled correctly in an experiment using e.g. a corpus of newswire texts, it can be assumed that official language standards (along with its rules that limit the creativity in creating new words by e.g. morphological processes such as compounding [112]) deliver a more or less clear cut instrument for judging the correct spelling of a word.
Our Survey collections show that in communities language is used in a far more liberal way. On the one hand, as explained in chapter 1, we find that text-based communication in communities often emphasizes higher levels of synchronicity than e.g. texts in a newspaper. Examples are chats, instant messengers and discussion boards which often have an answer latency of less than a few seconds to a few minutes. Also due to the dense net of social relations between community members, we find that language emphasizes these social relations between the communicating parties. Furthermore, the common pursuit of the community which often corresponds to a special interest necessarily results in the usage of a specialized community vocabulary.
Higher **synchronicity** may lead to text simulations of spoken language like onomatopoetic particles like "uhm" or "aaahh" or shortening of words like in "hangin around". Social relations may lead to phenomena where the content of the interest-sets is deliberately semantically blurred to e.g. express a certain attitude towards the community and its community support services. An example from the MTV community is "fo shizzle my nizzle bizzle dizzle" which might be complete nonsense intended to communicate a cynical or funny rejection attitude towards explicitly stating ones interests in this community.
In terms of **specialized language**, the MTV community collection also contains many names of bands and artists and even special skater vocabulary like "fs bigspin; kickflip; heelflip; 360 flip" which denote special moves.
Respecting these special forms of expression we judged the correctness of a spelling of a word

	Survey Collecion	MTV Collection	Party Collection
Number of Words	1630	2070	532
Number of "Real Errors"	25	12	5
Number of "Marked as Error"-Words in Microsoft Word	108	212	34
"Real" Error Ratio r_{real}	0.015	0.006	0.009
Microsoft Error Ratio $r_{\text{Microsoft}}$	0.066	0.102	0.064
Ratio $r_{\text{Microsoft}}$ / r_{real}	4.4	17.0	7.1

Table 4.1: Spelling errors in the interest test collections.

subjectively by leaving out deliberately incorrect spellings, special vocabulary and upper-case / lower-case errors. What was counted were those miss-spellings that were judged to be **not intended by the user**. We compared these figures against a state of the art spell-checker (included in Microsoft Word). Table 4.1 shows the results. It is quite obvious and no surprise that the "real" error rates are much higher for the survey collection which was conducted with non-native English speaking persons than for the other two collections which originate from the Anglo-American language area. More interesting than that is the ratio $r_{\text{Microsoft}}$ / r_{real}

which is a measure of how special the vocabulary of the respective community is. In the survey collection which was collected in a rather artificial experiment, the vocabulary is quite common: Microsoft Word detects only about four times as much errors as were judged to be "real". This is different for the MTV collection which subjectively contains the most specific vocabulary, the most "slang" writings and spoken-language-particles which is supported by the fact that MS Word detects 17 times as much errors as were judged to be "real" (unintentional).
These figures and the overall analysis of the test collections and the analysis of the difficulties in constructing a spelling checker with modest expenses and nevertheless near-perfect accuracy suggest that an adoption of automatic spelling correction for the processing of free text interest phrases is **not appropriate**. Even if an excellent state of the art spelling correction algorithm would be available, the creativity of the community members in creating a highly specialized vocabulary would render this correction algorithm useless, since isolated non-word errors cannot easily be distinguished from intended miss-spellings, weird new words and names, comic strip language and the like. Each of these problems can be addressed with appropriate heuristics, extensive community specific lexica and context sensitive spelling correction algorithms but the effort to do so would far outweigh the disadvantages of missing a small number of miss-spelled words in the calculation of similarity.

4.7.2 Comparing Unordered Sets of Interest Phrases

What can truly be said of the **space of possible interest concepts constructable by free text phrases** is that is **extremely large**. In contrast to the finite set of alternatives in the list-of-choice case, natural language allows to freely combine sub-concepts, qualifiers and other semantic elements and syntactic parts of speech. Even the number of word-compounds that can be constructed is, in principle, infinite. Certainly, rules of common sense may somehow limit the space of possible concepts. It may certainly also be possible to technically enumerate all (syntactically and semantically correct and incorrect) possibilities for such phrases by simply enumerating all combinations of words of arbitrary length over a finite alphabet with the diagonal method. But even for the subset of syntactically correct phrases no enumeration is known not to mention an enumeration for syntactically and semantically correct phrases. (For both we would first have to state what "syntactically correct" and "semantically correct" exactly means, which is also impossible).
As a consequence we can practically regard the set of possible interest phrases to be **infinite** and also regard it as a set which **practically cannot be enumerated**. Therefore we cannot speak of vectors or lists of free text interest phrases but have to speak of sets of free text interest phrases.
Seeking for an analogon for the simple expression 4.2 for comparing list-of-choice vectors, where the vector elements are pair-wise (dimension-wise) compared, implies that we will have to **compare each concept** of one set **with each concept** of the other set. What has to be regarded when constructing a similarity measure is that the similarity between two identical sets must be equal to one [109].
We denote the two sets of concepts (each described by a free text interest phrase) as $\mathcal{X}^{(j_1)} = \{C_0^{(j_1)}, C_1^{(j_1)}, \ldots, C_{|\mathcal{X}^{(j_1)}|-1}^{(j_1)}\}$ and $\mathcal{X}^{(j_2)} = \{C_0^{(j_2)}, C_1^{(j_2)}, \ldots, C_{|\mathcal{X}^{(j_2)}|-1}^{(j_2)}\}$. Assuming that we al-

ready have a similarity measure for single concepts, we can compute a **similarity measure**

$$\begin{aligned}\mathrm{sim}(\mathcal{X}^{(j_1)},\mathcal{X}^{(j_2)}) &= \frac{1}{2}\frac{1}{|\mathcal{X}^{(j_1)}|}\sum_{i=0}^{|\mathcal{X}^{(j_1)}|-1}\max_k \mathrm{sim}(C_i^{(j_1)},C_k^{(j_2)}) \\ &\quad + \frac{1}{2}\frac{1}{|\mathcal{X}^{(j_2)}|}\sum_{k=0}^{|\mathcal{X}^{(j_2)}|-1}\max_i \mathrm{sim}(C_i^{(j_1)},C_k^{(j_2)}) \quad . \end{aligned} \tag{4.5}$$

That means that we start from the first set and for each of its concept we compare this concept to each of the concepts in the other set, determine the concept of the other set which gives the highest similarity value and add this value to the value of the sum. We repeat this procedure starting with the second set and compute the average of both computations. This is necessary, since both sets can be of different extension. The heuristic behind 4.5 is that we compare a concept to that concept of the other set which is the best "candidate" for comparison. The best candidate is determined by choosing the one with the highest similarity.
If we compare **identical sets** we get a similarity value of one, which is what is desired:

$$\begin{aligned}\mathrm{sim}(\mathcal{X}^{(j_1)},\mathcal{X}^{(j_1)}) &= \frac{1}{2}\frac{1}{|\mathcal{X}^{(j_1)}|}\sum_{i=0}^{|\mathcal{X}^{(j_1)}|-1}\max_k \mathrm{sim}(C_i^{(j_1)},C_k^{(j_1)}) \\ &\quad + \frac{1}{2}\frac{1}{|\mathcal{X}^{(j_1)}|}\sum_{i=0}^{|\mathcal{X}^{(j_1)}|-1}\max_k \mathrm{sim}(C_k^{(j_1)},C_i^{(j_1)}) \\ &= \frac{1}{2}\frac{1}{|\mathcal{X}^{(j_1)}|}\sum_{i=0}^{|\mathcal{X}^{(j_1)}|-1}\mathrm{sim}(C_i^{(j_1)},C_i^{(j_1)}) \\ &\quad + \frac{1}{2}\frac{1}{|\mathcal{X}^{(j_1)}|}\sum_{i=0}^{|\mathcal{X}^{(j_1)}|-1}\mathrm{sim}(C_i^{(j_1)},C_i^{(j_1)}) \\ &= \frac{1}{2}\frac{1}{|\mathcal{X}^{(j_1)}|}\sum_{i=0}^{|\mathcal{X}^{(j_1)}|-1}1 \;+\; \frac{1}{2}\frac{1}{|\mathcal{X}^{(j_1)}|}\sum_{i=0}^{|\mathcal{X}^{(j_1)}|-1}1 \\ &= 1 \quad . \end{aligned} \tag{4.6}$$

In case of list-of-choice interest vectors, the dimensions of the vector space clearly and trivially determine (in equation 4.2) which concepts of one vector are to be compared with which concepts from the other vector. Since we have a clear enumeration of the dimensions (possible interest concepts (choices)) we can easily transform equation 4.2 for the list-of-choice similarity measure sim^{l-o-c} into a form closely resembling the more general form of 4.5.
Let the selectable concepts be enumerated $C_0, C_1, \ldots C_{|\mathcal{V}|-1}$. The binary list-of-choice vectors $x^{(j_1)}$ and $x^{(j_2)}$ can be written as sets:
$\tilde{x}^{(j_1)} = \{C_{i_0}, C_{i_1}, \ldots, C_{i_{|x^{(j_1)}|-1}}\}$ and $\tilde{x}^{(j_2)} = \{C_{k_0}, C_{k_1}, \ldots, C_{k_{|x^{(j_2)}|-1}}\}$.

Without the normalization factor $1/||x^{(j_1)}||\ ||x^{(j_2)}||$ of equation 4.2 we have then

$$\begin{aligned}
\mathrm{sim}^{l-o-c}(x^{(j_1)}, x^{(j_2)}) &\sim \sum_{i=0}^{|\mathcal{V}|-1} x_i^{(j_1)} x_i^{(j_2)} \\
&= \frac{1}{2} \sum_{i=0}^{|\mathcal{V}|-1} x_i^{(j_1)} x_i^{(j_2)} + \frac{1}{2} \sum_{i=0}^{|\mathcal{V}|-1} x_i^{(j_1)} x_i^{(j_2)} \\
&= \frac{1}{2} \sum_{m=0}^{|\tilde{x}^{(j_1)}|-1} \max_{k_n} \delta_{i_m k_n} + \frac{1}{2} \sum_{n=0}^{|\tilde{x}^{(j_2)}|-1} \max_{i_m} \delta_{i_m k_n} \\
&= \frac{1}{2} \sum_{m=0}^{|\tilde{x}^{(j_1)}|-1} \max_{k_n} \mathrm{sim}(C_{i_m}^{(j_1)}, C_{k_n}^{(j_2)}) \\
&\quad + \frac{1}{2} \sum_{n=0}^{|\tilde{x}^{(j_2)}|-1} \max_{i_m} \mathrm{sim}(C_{i_m}^{(j_1)}, C_{k_n}^{(j_2)})
\end{aligned} \tag{4.7}$$

using the Kronecker symbol

$$\delta_{xy} = \begin{cases} 1 & \text{if} \quad x = y \\ 0 & \text{if} \quad x \neq y \end{cases} . \tag{4.8}$$

This consideration shows that in case of binary list-of-choice vectors (without consideration of *idf* measure or taxonomic structure) we implicitly assume a trivial similarity measure $\mathrm{sim}(C_{i_m}^{(j_1)}, C_{k_n}^{(j_2)}) = \delta_{i_m k_n}$ that considers two concepts as similar if they are equal and as dissimilar if they are not equal.
If we consider *idf* measure and taxonomic structure as shown in figure 4.5 we have to substitute a slightly more complicated similarity measure for two concepts but we can still "fit" the list-of-choice vector similarity measure with the overall form of 4.5. This shows that the list-of-choice case is closely related to the free text phrase case. What is much more difficult in the free text phrase case is to find an appropriate similarity measure between two concepts $\mathrm{sim}(C_i^{(j_1)}, C_k^{(j_2)})$ which we will now discuss.

4.7.3 Conceptual Semantic Relatedness

In our previous considerations with respect to list-of-choice vectors we have built a similarity measure on the basis of a **taxonomy (simple ontology) of choice alternatives** with generalized abstraction relations. The introduction of generalized abstraction relations was mainly motivated by the fact that the semantics of the ontology and especially the precise semantics of its relation edges is considered to be a **given community agreement**. The community decides about the semantics and will fine tune it to its needs. The similarity measure between two concepts from this ontology that was proposed in figure 4.5 is not sensitive to the exact type of general abstraction relation employed in the ontology and is very simple (see previous section). Since the ontology is given and finite, the similarity between two concepts within the ontology can essentially also be assumed to be a given quantity which a community can fine tune with respect to its needs.
In case of concepts described by **free text** phrases, the **ontology** is not explicitly given but

is **implicitly determined and presumed** by the community member's knowledge, cultural background, language abilities and preferences etc.. Thus we will have to be more precise in terms of relations between concepts. The notions **semantically related, semantically similar** and **semantically distant** have to be distinguished. (As has been stated before, the term concept is extensionally defined and embraces classes as well as instances.)
Intuitively, the concepts "train" and "conductor" are semantically more closely related than the concepts "train" and "truck" although "train" and "truck" can be considered more similar than "train" and "conductor". How can we define these terms more precisely? A rather narrow definition of similarity is "to have many properties in common". In the upper example, "train" and "truck" for example have the property "chassis-material" in common ("steel"). This corresponds to a Frame-like view. In essence, this notion of similarity can be based on "property-of" and "value-of-property" semantic relations. In [18, 19] it is emphasized that the more natural term in comparing concepts is semantic relatedness (where the type of relation has to be stated) with semantic distance as its opposite. It is nevertheless also emphasized that the notion of similarity is much more widespread.
For our purposes the clearest view is that semantic relatedness is the more basic notion while **similarity** is a meta-notion that **builds upon one or more semantic relations**. We distinguish between the various standard types of semantic relations (such as "is-a", "part-of", "property-of" etc.) which we will treat on the same level and will not sub-divide in core and schema relations like it is the case in models like RDF and RDF-Schema.

4.7.3.1 General Tools and Techniques from NLP

The concepts (elements) of the interest sets are described in natural language phrases. So the **Semantics and Pragmatics branches of Natural Language Processing (NLP)** deliver the basic tools and techniques to analyse semantics relations between these concepts and construct a similarity measure which is meaningful with respect to the goals of this thesis (that is identifying and analyzing Ad-Hoc Groups in mobile communities and providing improved information management community services to those groups). Many sub-branches of NLP contribute to the Semantics and Pragmatics branch: **Morphology** investigates the form of words in a grammatical context. The form of words is changed via morphological processes like **inflection** (changing its tense, plurality etc. by means of adding prefixes, suffixes etc. to a root form), **compounding** (combining two or more words into a new word) or **derivation** (often changing meaning or function of a word) [112]. **Syntax, Part-of-Speech-Tagging and Parsing** deal with the construction, analysis and application of formal grammars for a natural language. Finally the construction of **lexical databases** deals with the automated or hand-crafted collection of databases like **dictionaries, thesauri, semantic nets** etc. which contain large amounts of semantic and syntactic facts about words, collocations (two-word-phrases) etc. [38] in various degrees of formalization. The term lexicon itself is not precisely defined throughout NLP literature. In [112] lexicon is defined as the part of a grammar's rules which substitute a word of a given category for that category's symbol whereas in [38] lexicon is more or less liberally used as a synonym for lexical database.
In [38] it is stressed that often a distinction is made between **word-knowledge** and **world-knowledge**. Word-Knowledge databases can be classified as dictionaries while world-knowledge databases can be classified as encyclopedias [38]. The term database embraces electronic forms as well as printed forms (books).
Dictionaries like Logman Dictionary of Contemporary English (see [18]) contain paraphrases

for each sense of a word and may contain syntactic and grammatical information about how to spell or use the word in a sentence. This type of information is referred to as word-knowledge.
Encyclopedias explain concepts which are identified by words. Usually for each such concept a short text informs about this concept, establishes relationships with other concepts by adding background information that allows to embed the information into the context of already acquired world-knowledge. Naturally the boundaries between the two forms of knowledge and their corresponding databases are fuzzy [38] because word knowledge and world knowledge cannot be precisely distinguished.
Another resource for semantic relations between words (word-knowledge) are **Thesauri** which in their most basic form "group words by idea" [121](in [18]), that is sets of synonymous or semantically very closely related words are enumerated and an index for effective searching is provided.

For the goals of this thesis, a mere **statistical approach** that completely neglects analytical aspects of NLP like e.g. parts of speech and that does not make use of semantic lexical resources such as semantic nets does not seem very promising since the sets of free text interest phrases do not contain very many words and so the statistical basis will, in general, be too small.
On the other hand, an approach that uses very **fine grained NLP analytics** like fine grained part-of-speech tagging, exact morphological analysis and parsing with complicated grammars is also not very appropriate. This is mainly due to pragmatic reasons, because required resources like grammars and reliable tools are difficult to obtain, require extensive experience in handling or do not perform with the required accuracy and reliability. Furthermore, there is evidence from other fields of applications of NLP such as machine translation, that rational rule-based methods are already outperformed or will sooner or later be outperformed by statistical or hybrid approaches [147].
Therefore for our purposes, a heuristic based hybrid approach using semantic lexical resources seems more appropriate than a completely statistical or an extensively analytic approach. We will now investigate, why lexical resources that have a semantic net architecture like WordNet are an appropriate tool for this approach.

4.7.3.2 WordNet as an Example for a Semantic Net

WordNet is a semi-formal ontology, lexical database or semantic net developed mostly for natural language processing tasks [38], [43], [117]. It tries to define the aspects of the semantics of English vocabulary (word-knowledge), by providing for every word a pair (f, s) where f is the **word-form** and s the **word-semantics**. f is a string and s is defined via a set of synonyms for the word which define its meaning. These sets are called **Synsets**.

WordNet subdivides words into four syntactic categories: **nouns, verbs, adjectives and adverbs**. Besides the syntactic categories (which form a separate conceptualization), the main representation for concepts are the Synsets. The concepts are extensionally defined by enumerating synonymous words. WordNet contains about 90000 Synsets [117]. (The Synsets are not disjoint, which means that a word-form can have multiple meanings (be an element of more than one synset)(Polysemy)). Those 90000 concepts are linked by **binary semantic relations** called pointers which relate every element of the first Synset to an element of the second. These relations are fixed. Some of them are [18]:

- Hyponymy ("is-a"). Relation between noun Synsets.

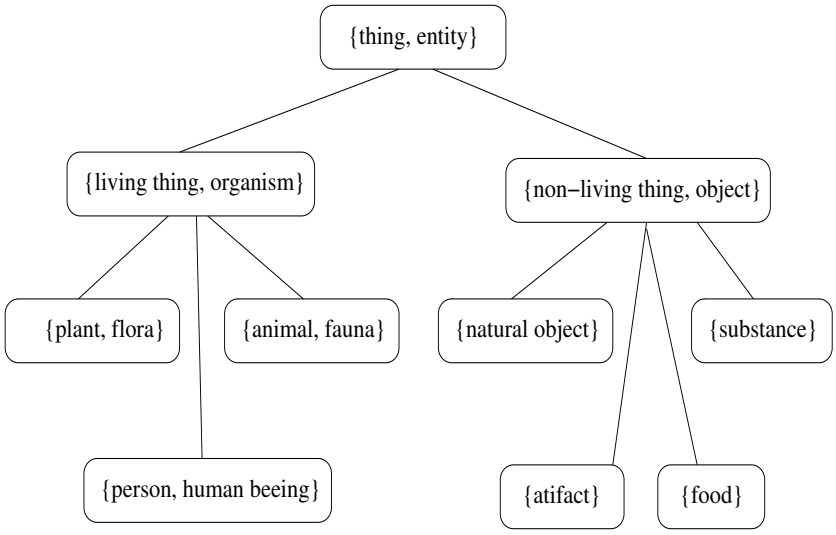


Figure 4.7: ([43]) Small part of WordNet showing is-a relations among concepts defined as Synsets of nouns (here a relatively abstract part of tangible things).

- Hypernymy ("subsumes") Inverse "is-a".
- Meronymy ("part of"). In fact WordNet distinguishes three types of Meronymy relations:
  - PART_MERONYM "component of" ("wheel"-"car")
  - MEMBER_MERONYM "element-of" ("ship"-"fleet") or
  - SUBSTANCE_MERONYM "material property-of" ("steel"-"sword").
- Holonymy ("has-a") Inverse "part-of". (In the same three variants).

WordNet distinguishes semantic relations that hold between Synsets (**conceptual semantic relations**) and semantic relations that hold between words only (**lexical semantic relations** (in the narrower sense). An example is the Antonymy ("opposite of") relation between pairs of adjectives and adverbs (Example: ("wet","dry")).

WordNet does not have axioms nor an inference mechanism. Concepts and relations are represented **extensionally**. Since some authors put a special emphasis on the role of **intensional** logic-based elements (such as axioms) as integral parts of ontologies it can be questioned, whether WordNet is an ontology at all.
Furthermore, WordNet **does not contain world-knowledge** like an encyclopedia and its concept-descriptions are not structured further. Other ontologies, which are more strongly influenced by Frame Logic or object-oriented paradigms have slots and properties for concepts (classes, objects) as description elements, which in turn have possible value ranges. These description elements could, in principle, be described by relations too, as has been mentioned earlier but this is not implemented in WordNet.

WordNet has some main characteristics of a **Thesaurus** because it defines a word's meaning by providing synonyms for it. It also far exceeds a thesaurus because it contains explicit semantic relations and also contains super-concepts which are described by phrases instead of words. (A thesaurus only contains lexicalized concepts (that are described by single words) [38]). These

concepts are necessary for a reasonable structuring of the semantic net. An example that is given in [38] is the concept described by the single element Synset { "bad person" }. These concepts are introduced although there is no single word for it in the respective language (in our case English).
WordNet also contains elements of a **dictionary** because it gives definitions and sample sentences for most Synsets [38]. Besides that, it contains morphological data like e.g. a link between "behavioral" and "behavior" [38].
A major drawback of WordNet is that is does not contain relations that would indicate a shared membership of words (especially words from the four different syntactic categories (nouns, verbs etc.)) in a specific topic of discourse [38]. Examples are "reading" and "book" or "net", "racket" and "ball" (this is why this problem in [38] is referred to as the "Tennis Problem"). There have been some attempts to augment WordNet with such topical (or "domain"-)information (like e.g. [111]) but these attempts are too coarse grained to be really helpful in our case.
Nevertheless, WordNet is the only lexical resource that unites so many useful features for analyzing semantic relatedness while having such a large coverage of modern English.

4.7.3.3 Techniques for Measuring Semantic Relatedness with WordNet

For measuring the semantic relatedness, semantic similarity or semantic distance (all three will be summarily denoted with the term "semantic association") of concepts that are described with single words, numerous approaches exist in literature which we will now give some examples of.
A first class of methods relies on the heuristic of measuring semantic associations by **edge counting** in the semantic network. Simple edge counting would result in measuring the distance between Synsets C_1 and C_2 as the path-length of Hyponymy ("is-a") or Hypernymy ("subsumes" (the inverse relation)) relations in the semantic net that connects the two Synsets (see [18, 19]). In [71] a slightly more differentiated approach is suggested. (see [18, 19]). The authors distinguish essentially between two non-trivial strength levels between words. The strong level is assumed if the words are contained in the same Synset or are antonyms or one word is contained as a compound in the other word and both Synsets are linked by a path of length one. The medium strong level is assumed if the Synsets of both words are linked by a path of length < 5 which is of one of eight types described in the paper. The semantic distance between the two words is then proportional to the sum of path length and changes of direction of this path.
A second class of techniques takes into account that short path lengths between two concepts in a part of the taxonomy of "is-a" relations which is closer to the leafs is more meaningful with respect to similarity than a short path between two concepts in a part of the is-a-taxonomy which is closer to the root (see [18]. This observation is motivated by an increasing "conceptual density" in the direction from the root to the leafs. One example for this approach is given in [197]:

$$\mathrm{sim}(C_1, C_2) = \frac{2\, n_3}{n_1 + n_2 + 2\, n_3} \tag{4.9}$$

where n_1 and n_2 are shortest path lengths from concepts C_1 and C_2 to their least common super-concept C_3 and n_3 is the path length from C_3 to the root of the hierarchy (see also [18]). The incorporation of n_3 emphasizes the increasing conceptual density towards the leafs of the taxonomy.
Other approaches like [157] combine information theory based on probabilities that were won through corpus statistics with taxonomic information from the semantic net. His approach

gives the similarity between two concepts $\mathrm{sim}(C_1, C_2)$ as the information of their most specific super-concept $C = \mathrm{mss}(C_1, C_2)$) in the taxonomy (see also [18]):

$$\mathrm{sim}(C_1, C_2) = -\log\ p(C) = -\log\ p(\mathrm{mss}(C_1, C_2)) \tag{4.10}$$

The probability $p(C)$ is the probability that an instance of the concept C is encountered in the corpus.
This type of approach seems very promising since a very similar technique presented in [79] was evaluated as the best overall similarity measure in a comparing study of five similarity measures using WordNet [19].
For our purposes, no corpus with enough statistical relevance to use methods of this type is available, so we need to develop an adapted measure that respects the special features of free text interest phrases best. Before we can do this, a last aspect needs to be discussed which will complete our arguments for the choice of our measure.

4.7.4 Disambiguating Word Senses

In WordNet, 12 per cent of the words are polysemous [105], that is they have several different senses. An example is the word "bank" among the senses of which are a sense close to financial institute and also a sense close to shore, land formation. Deciding which sense of a word applies in a given textual context is referred to as **Word Sense Disambiguation** (WSD) [105, 112].
Algorithms for WSD treat the problem as a classification problem: A word w and its context $\chi(w)$ in a text must be mapped to a sense s from a set of senses $\mathcal{S}(w)$ of the word:

$$(w, \chi(w)) \overset{\mathrm{WSD}}{\longrightarrow} s \in \mathcal{S}(w) \tag{4.11}$$

Many possibilities exist for defining the context $\chi(w)$ of a word. Usually **local context** and **topical context** are distinguished [105]. Topical context is computed from the words (mostly substantives) that occur immediately before a word and immediately after a word. The number of words that are taken into account is often called the context-**window** [112]. Typical window sizes are 2 to 25 words before and after the word's occurrence (see [105]). Local context consists of syntactical and semantic features of the sentence(s) that the word is part of. An example of local context are part-of-speech patterns which can be generated by a part of speech tagger.
Although classifiers perform "almost perfect" [105] on non-related word senses (e.g. the two senses in the upper example "bank") and still reasonably well on closely related word senses (e.g. "bass" (the voice) and "bass" (the instrument-class)) when combined with other measures [105], there is one big problem with WSD-classifiers: the **lack of training data**. Almost all scientific projects referred to in papers about WSD had to manually compile large sets of training data

$$\Big\{\big(w_i, \chi_1(w_i), s(w_i, \chi_1(w_i))\big), \big(w_i, \chi_2(w_i), s(w_i, \chi_2(w_i))\big), \ldots, \big(w_i, \chi_n(w_i), s(w_i, \chi_n(w_i))\big)\Big\}$$

for the sense tagged contexts of one word w_i in their experiments. The number of required training set elements varies with the sort of context used but can be assumed to be order of magnitude $n = 100$ to deliver a reasonable classification accuracy (order of magnitude $80 - 90\%$) [105]. Since it is very costly to manually compile such a collection for a reasonable part of the English vocabulary, such a (standard) compilation does not exist yet.
The consequence of this is that we **cannot perform usual classifier based WSD** on a set

of free text interest phrases because no such training data is available. Furthermore, special community vocabulary or special community senses for common words will most likely substantially deteriorate the performance of conventional WSD strategies (trivial examples are "fat" or "cool").
We will therefore have to rely on local semantic contexts (if present) which will be represented with the a semantic net to perform some basic WSD which we will now discuss in connection with our similarity measure.

4.7.5 Similarity Measure for Pairs of Free Text Interest Phrases

Subsuming what has been said in the previous subsections we find that almost all conventional NLP approaches cannot be applied or seem less favorable in case of free text interest phrases:

- **Spelling Corrections** are not well applicable because of special community vocabulary
- **Syntactic and Grammatical Analysis** is complicated to perform, needs a lot of resources and is not very useful without large scale datasets that allow to link the results of such an analysis with semantic aspects (like word senses). Furthermore, free text interest phrases are not complete sentences in most cases which would require adaptations to the available tools.
- Conventional classifier based **WSD** is not applicable because of the lack of training data and the special community vocabulary.
- Simple **Thesauri** lack sophisticated semantic relations among Synsets.
- **Statistical approaches** cannot be applied (at least not in the usual fashion) because free text interest phrases contain too few statistical information.

Thus we will rely on approaches based on **Semantic Nets** which are the most developed form of lexical resources with respect to semantics on the level of word-knowledge. Of the semantic nets freely available, **WordNet** is the most developed resource.
Thus we will propose the following WordNet-based procedure for a similarity measure for pairs of free text interest phrases:

Let the interest concepts $C_i^{(j_1)}$ and $C_k^{(j_2)}$ from the interest sets

$$\begin{aligned} \mathcal{X}^{(j_1)} &= \{C_0^{(j_1)}, C_1^{(j_1)}, \ldots, C_{|\mathcal{X}^{(j_1)}|-1}^{(j_1)}\} \\ \text{and} \quad \mathcal{X}^{(j_2)} &= \{C_0^{(j_2)}, C_1^{(j_2)}, \ldots, C_{|\mathcal{X}^{(j_2)}|-1}^{(j_2)}\} \end{aligned}$$

be textually described by the free text interest phrases $\hat{C}_i^{(j_1)}$ and $\hat{C}_k^{(j_2)}$

$$\begin{aligned} C_i^{(j_1)} &\hat{=} \hat{C}_i^{(j_1)} = (w_{n_0}, w_{n_1}, \ldots, w_{n_{N(\hat{C}_i^{(j_1)})-1}}) \\ C_k^{(j_2)} &\hat{=} \hat{C}_k^{(j_2)} = (w_{m_0}, w_{m_1}, \ldots, w_{m_{N(\hat{C}_k^{(j_2)})-1}}) \end{aligned}$$

where the function $N(\hat{C})$ counts the number of words in a free text interest phrase $\hat{C}$ and words are represented by the symbol w. For a convenient formulation will assume in this equation and

all following equations of this section an enumeration (global identification index) of the words contained in all free text interest phrases in all sets of the collection. (E.g. in the above equation this results in using the indices $n_0, n_1, \ldots$ and $m_0, m_1, \ldots$). The set of all the words contained in all free text interest phrases in all sets $\mathcal{X}^{(j)}$ of the collection is called the vocabulary $\mathcal{V}^{\text{coll}}$ of the collection: $\mathcal{V}^{\text{coll}} = \bigcup_j \mathcal{V}(\mathcal{X}^{(j)})$.
From the phrases, the **stop-words** are removed with the help of a stop-word list. For our purposes e.g. the stop-word list [165] compiled by the reputable researchers Salton and Buckley in connection with their mid-1990s SMART project on information retrieval is suitable. Stop-words are words like "is", "can", "has" that are believed to carry little semantics in a given topical context (in a text fragment about waste management the word "can" may carry substantial semantics !). If no context is specified, the stop-word-list is assumed to apply to all contexts.
Stop-word-removal is mainly used in information retrieval. The reason for using it here is partly due to the special vocabulary used in a community setting and partly due to the nature of our approach: If we could assume that all verbs, nouns, adjectives and adverbs used in the free text interest phrases of the community were represented in WordNet, we could drop all the other word forms, if we assumed that they do not contribute much to the phrase's semantics. It must be emphasized that the validity of this assumption would only have to be considered in connection with the approach described here (see below). If the proposed approach would include a detailed grammatical or part of speech analysis, it would not be valid to drop e.g. pronouns or other word types ("teaching myself English" is surely a different interest concept than "teaching others English").
Unfortunately, as the discussion in previous sections shows, the community vocabulary may be so special, that a substantial part of the constituting words in the phrasal descriptions of the interest concepts are not even represented in current spelling checker databases let alone in WordNet. This renders us with the situation that we cannot simply drop everything that is not represented in WordNet. Even systematically ruling out other word forms except verbs, adjectives, adverbs and nouns is difficult because NLP tools for this purpose will have severe problems with special community vocabulary as well. We therefore rely on simple stop-word removal which, in short words, corresponds to the positivistic approach of removing words which we explicitly assume to be irrelevant instead of neglecting all the words we do not automatically know.
Furthermore, we will **delete all multiple instances** of words from the phrases should this case occur which is not likely.
The free text interest phrases after stop-word removal (SR) and multiple instance removal (MIR) will be denoted by $\widetilde{C}$:

$$\begin{aligned} \hat{C}_i^{(j_1)} \xrightarrow{\text{SR, MIR}} \widetilde{C}_i^{(j_1)} &= (w_{p_0}, w_{p_1}, \ldots, w_{p_{N(\widetilde{C}_i^{(j_1)})-1}}) \\ \hat{C}_k^{(j_2)} \xrightarrow{\text{SR, MIR}} \widetilde{C}_k^{(j_2)} &= (w_{q_0}, w_{q_1}, \ldots, w_{q_{N(\widetilde{C}_k^{(j_2)})-1}}) \end{aligned}$$

The basic idea is now analogous to the approach described in 4.7.2: Assuming that we have a similarity measure for single-word-concepts, we will symmetrically compare each word from the

one phrase with each word each word from the other phrase and choose the best match:

$$\begin{aligned}\mathrm{sim}(C_i^{(j_1)}, C_k^{(j_2)}) &= \mathrm{sim}(\widetilde{C}_i^{(j_1)}, \widetilde{C}_k^{(j_2)}) \\ &= \frac{1}{2}\,\frac{1}{N(\widetilde{C}_i^{(j_1)})} \sum_{a=0}^{N(\widetilde{C}_i^{(j_1)})-1} \max_b \mathrm{sim}(w_{p_a}, w_{q_b}) \\ &\quad + \frac{1}{2}\,\frac{1}{N(\widetilde{C}_k^{(j_2)})} \sum_{b=0}^{N(\widetilde{C}_k^{(j_2)})-1} \max_a \mathrm{sim}(w_{p_a}, w_{q_b}) \quad . \end{aligned} \tag{4.12}$$

This corresponds to modeling the relations between the semantics of concepts described by free text phrases and their constituting lexicalized concepts (single word concepts) in a simple way which nevertheless seems justified considering the previous arguments.

In order to compute the similarity between lexicalized concepts we use the word-knowledge from WordNet. We will associate a **lexicalized concept** which is described by a single word w_{r_1} with a **synset** $S_t = \{w_{r_1}, w_{r_2}, \ldots w_{r_{|S_t|}}\}$ in WordNet which contains the word and possibly several synonymous words.
For notational convenience, we assume an enumeration (global identification index) of all the Synsets contained in WordNet for each of the four word classes (verbs, nouns, adjectives, adverbs). Establishing a different enumeration for each word-class is reasonable, because there are no relations in WordNet between Synsets from different word-classes. Furthermore, we add an abstract root node Synset S_0 for each the four word-classes which is a common practice [105]. From the various relations we will regard Hyponymy ("is-a") and Meronymy ("part-of") relations. From the three variants of Meronymy in WordNet we consider only PART_MERONYM and MEMBER_MERONYM relations and drop the SUBSTANCE_MERONYM relation because it is not very appropriate for our purpose. If a Synset is related to another synset via one of these relations we will, in analogy to section 4.6.2, denote this as a **generalized abstraction relation** (GAR).

$$\left.\begin{array}{l} S_{t_1} \xrightarrow{\text{is-a}} S_{t_2} \text{ (denoted by } H(S_{t_1}, S_{t_2})) \\ \text{or} \\ S_{t_1} \xrightarrow{\text{part-of}} S_{t_2} \text{ (denoted by } M(S_{t_1}, S_{t_2})) \end{array}\right\} \Rightarrow S_{t_1} \xrightarrow{\text{GAR}} S_{t_2} \text{ (denoted by } G(S_{t_1}, S_{t_2}))$$

From the semantics of "is-a" and "part-of" relations it follows (just like in section 4.6.2 that the directed graph that results from WordNet by regarding only GAR edges is cycle-free and thus the graph can also be regarded as tree-like (see section 4.6.2).
We will denote the shortest path length from Synset S_{t_1} to Synset S_{t_2} using only GAR edges by $pl(S_{t_1}, S_{t_2})$.
Having prepared these notations we can develop an expression for the **similarity measure between lexicalized concepts**:

$$\begin{aligned}\mathrm{sim}(w_{p_a}, w_{q_b}) &= \big(1 + \mathrm{dist}(w_{p_a}, w_{q_b})\big)^{-1} \\ &= \begin{cases} 1 & \text{if} \quad w_{p_a} = w_{q_b} \\ \left(1 + \min_{\{(c,d)|w_{p_a} \in S_{t_c} \wedge w_{q_b} \in S_{t_d}\}} \left(\frac{pl(S_{t_c}, S_t^{\mathrm{ncsc}}) + pl(S_{t_d}, S_t^{\mathrm{ncsc}})}{2\, pl(S_t^{\mathrm{ncsc}}, S_0)^{\frac{1}{2}}}\right)\right)^{-1} & \text{if} \quad w_{p_a}, w_{q_b} \in \text{WordNet} \\ 0 & \text{else} \end{cases} \end{aligned} \tag{4.13}$$

If not both words are contained in WordNet (which is denoted by $w_{p_a}, w_{q_b} \in$ WordNet in this equation) we can only compare the words with respect to identity.
This approach is closely related to the approaches described in 4.7.3.3 and especially related to the approach proposed in [197]. The principle idea is that the distance between two lexicalized concepts is proportional to the average (arithmetic mean) of the path-lengths of these nodes to its nearest common generalized super-concept (a super-concept with respect to GARs). This super-concept is denoted by S_t^{ncsc} in the equation. The distance is furthermore inversely proportional to the "specifity" of S_t^{ncsc}, that is how deep down in the taxonomy this common concept is rooted. The nearer to the leafs, the more specific is the concept and the nearer to the root the more general is the concept.

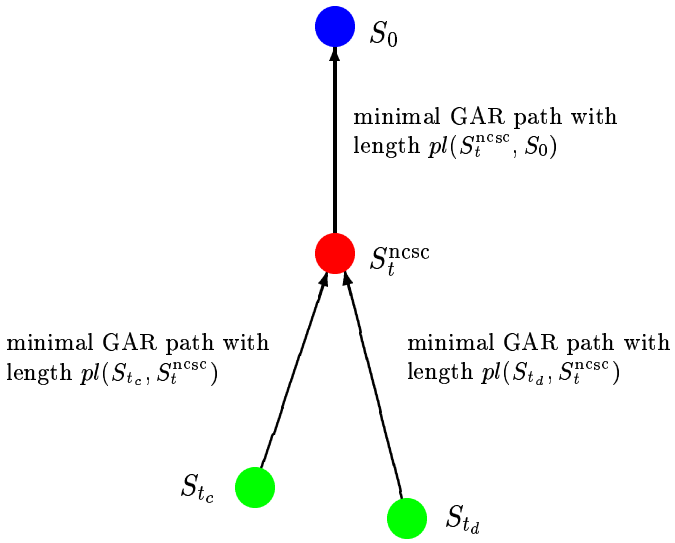


Figure 4.8: Path lengths between Synsets, their nearest common generalized super-concept and the root node

The similarity expression differs in four points from the approach proposed in [197] given in equation (4.9):

- The expression contains a minimization about all the Synsets (senses) that a word is in while in [197] a fixed (predetermined) word-sense is assumed. This minimization is an optimistic approach which is due to the lack of a proper WSD.
- We count edges instead of nodes. If the two words are in the same Synset, the path-lengths $pl(S_{t_c}, S_t^{\text{ncsc}})$ and $pl(S_{t_d}, S_t^{\text{ncsc}})$ are equal to zero. If $S_t^{\text{ncsc}} = S_0$, the distance is infinite and the similarity is equal to zero.
- We consider Meronymy relations as well as Hyponymy relations for computing the path lengths
- The overall level of the nearest common super-concept (with respect to GAR) is less significantly featured in our equation by introducing a power of $1/2$ which is not present in [197]. This is justified by the observation that the influence of the depth of the generalized super-concept should be less significant than the influence of the averaged minimal path length of the two Synsets in question. E.g. a double increase from in $pl(S_t^{\text{ncsc}}, S_0)$ would

double the distance between the concepts which seems inappropriate for our purposes. Two related general interests may be less similar than two related special interests but it is questionable whether the difference in similarity should be that significant. Thus a more appropriate exponent of 1/2 was chosen.

- In contrast to [197] we will have to deal with the case, that words that we want to compare are not contained in WordNet

4.7.5.1 The Tennis Problem Modification

As has been discussed before, WordNet does not contain world - or domain-knowledge. That is, there are no "topically related to" links between words or Synsets in WordNet. This problem was called the Tennis Problem in [105]. (see 4.7.3.2). In case of list-of-choice interest vectors, the Tennis problem is easily coped with by the fact that the community determines the taxonomy from which the interest vectors are chosen be the users. That includes the possibility to include "functionally-related-to" general abstraction relation edges into the taxonomy.
For the case of free text interest phrases our approach to deal with the tennis problem is the following: We will add a simple modification to the upper approach (4.13) for computing the similarity using a dictionary / encyclopedia. Retrieving a dictionary entry in Microsoft Encarta [137] for the word "tennis" delivers

> game with ball, rackets, and net: a game played on a rectangular court by two, or two pairs of, players with rackets who hit a ball back and forth over a net

which relates "tennis" to "ball", "rackets", "net" etc..
Technically, we need to consider the Encarta Dictionary together with the Encarta Encyclopedia, because many words that identify e.g. historic persons like Bach or Beethoven have no definition in the dictionary which contains mostly narrow definitions which often have a rather lexical character. But these missing words (or the corresponding concepts) usually have extensive entries in the Encyclopedia (which contains world-knowledge). Thus we will from now on understand by a dictionary entry the concatenation of its dictionary entry and its Encyclopedia entry.
Comparing two lexicalized concepts on the basis of such dictionary entries alone may not be sufficient. Nevertheless we can supplement the WordNet-based similarity expression with the help of such dictionary entries.
Because we omit word sense disambiguation as has been explained before, we need to consider all senses provided by the dictionary.

Using a straightforward vector model approach (see [51]) we can transform a dictionary entry $d(w)$ for a word w into a word-vector by stop-word-removal and subsequently counting term-frequencies (CTF) for the remaining words in $\widetilde{d}(w)$:

$$d(w) \xrightarrow{SR} \widetilde{d}(w) \xrightarrow{CTF} \check{d}(w) = (\nu_{u_0}, \nu_{u_1}, \ldots, \nu_{u_{|\mathscr{V}(\widetilde{d}(w))|-1}})$$

$(\mathscr{V}(\widetilde{d}(w))$ denotes the vocabulary (the set of different words in $\widetilde{d}(w)$). Just as for the overall vocabulary $\mathscr{V}^{\text{coll}}$ of all interest sets in the collection, we assume an enumeration (a global index) for all the words occuring in all dictionary entries of all the words in $\mathscr{V}^{\text{coll}}$. This vocabulary will

be denoted by

$$\mathcal{V}^{\text{dict}} = \bigcup_{w \in \mathcal{V}^{\text{coll}}} \mathcal{V}(\widetilde{d}(w)). \tag{4.14}$$

We can then compute a vector model based cosine similarity between two dictionary entry term frequency vectors $\check{d}(w_p)$ and $\check{d}(w_q)$:

$$\text{sim}^{\text{dict}}(w_p, w_q) = \begin{cases} 1 & \text{if} \quad w_p = w_q \\ \frac{1}{||\check{d}(w_p)|| ||\check{d}(w_q)||} \sum_{\{(a,b)|w_{u_a} \in \widetilde{d}(w_p) \wedge w_{l_b} \in \widetilde{d}(w_q)\}} \nu_{u_a} \nu_{l_b} \, \delta_{u_a l_b} & \text{if} \quad w_p, w_q \in \text{Dictionary} \\ 0 & \text{else} \end{cases} \tag{4.15}$$

Using the tennis problem modification results in the equation for the **Overall Similarity Measure for Pairs of Free Text Interest Phrases**:

$$\text{sim}^{\text{overall}}(w_q, w_p) = \frac{1}{\alpha + \beta} \big(\alpha \, \text{sim}^{\text{WordNet}}(w_q, w_p) + \beta \, \text{sim}^{\text{dict}}(w_q, w_p) \big) \tag{4.16}$$

where $\text{sim}^{\text{WordNet}}(w_q, w_p)$ is the expression given in equation (4.13) and the parameters α and β need to be experimentally determined by the community.

4.7.5.2 A Remark on Inflections

Free text interest phrases contain words in all kinds of **inflectional forms** (plural, Genitive etc. for nouns; various tenses for verbs etc.) which are different from the basic forms (Nominative Singular for nouns, Infinitive for verbs etc.) of the words. So, since we have to look up all the words in the free text interest phrases in lexical databases (WordNet as well as in Encarta Dictionary), we would normally need basic forms of the words.
Among other reasons explained before, WordNet and Microsoft Encarta Dictionary were chosen, because they contain most inflected forms of the words in their database and **automatically resolve** the entry of the basic form when an inflected form is queried for. As an example, Encarta will give the entries for the different senses of "tree" if "tree" or "trees" is queried.
The process of retrieving a normal form can, in principle, be accomplished by simply additionally listing all inflected forms of a word in the database. More sophisticated approaches are applicable as well [112]. Note that the process of reducing words to their normal form is different from **stemming** which reduces a word to its word stem [112].

4.7.6 The Algorithm

Summarizing the bricks built in the previous sections we can now give the overall algorithm for the **Overall Similarity Measure for Sets of Free Text Interest Phrases**. The algorithm is shown in figures 4.9 and 4.10. All the parts of the algorithm and the underlying heuristics were discussed in the previous sections.

4.7.7 A Survey

Human similarity judgements between sets of free text interest phrases or pairs of list-of-choice interest vectors are influenced by a large number of factors which are extensively researched

in many fields of science. E.g. knowledge about **social structures** and their interaction with the stated interests will influence the decisions of human jurors. For example, we automatically associate a certain "prototype" of human being when somebody states {"surfing", "girls", "partying"} as his interests and a different prototype when interests like {"programming my computer", "listening to Kelly Family", "spending time with my mother"} are stated. Furthermore, different levels of knowledge about the **general areas of interest** reflected in the interest statements stated will influence the judgements as well (e.g. people above the age of 70 will certainly not be able to make a judgement about which of the interest pairs ("Korn", "Limp Bizkit") and ("Korn, Iron Maiden") is more similar, a judgement that will pose no problem whatsoever to most teenagers).
Although is very hard to precisely control and measure these influence factors in a scientifically accepted way, **comparison with human judgement** appears to be the only method to validate heuristic based similarity measures for measuring semantic relatedness or similarity between concepts described by text.
In [18] two other forms of validation of such measures are mentioned: Mathematical analysis and application specific evaluation. **Mathematical analysis** such as in [109] can only reveal certain general principles but will, in general, not be able to judge the quality of a given heuristic because no abstract model for human knowledge or language exists with which the validity of a given heuristic could be mathematically deduced. **Application specific analysis** is very difficult, because the similarity measures are typically used in applications (such as Ad-Hoc-Group Information Management) the quality of which can only be subjectively judged by the user. It is extremely hard to conduct controlled experiments which isolate and measure the influence of the similarity measure on the overall usefulness application.
Since because of the difficulties in controlling the influence factors, **no standardized test collections** of human judgements about the relatedness of e.g. lexicalized concepts exist (or can exist). Some authors (see [18]) use a very small test set of human judgements on similarities between lexicalized concepts which was collected in connection with the paper [118]. But this test set is not applicable for our purposes, because we need to compare interest phrases.
Thus a second **survey** (besides the survey that collected the "Survey Collection" (see 4.4.3) of free text interest phrases) was conducted using a selection from the interest test collections "Survey Collection" (see 4.4.3) and "Dating Collection" (see 4.3) and using a selection of single free text interest phrases from the "Survey Collection" which results in **3 separate surveys**.
30 test-persons from the personal social environment of the thesis author were emailed a questionnaire (see C). 10 persons were mailed the questionnaire for judging similarities between persons on the basis of sets of free text interest phrases from the "Survey Collection" (survey 1). 10 persons were mailed the questionnaire for judging similarities between persons on the basis of list-of-choice interest vectors from the "Dating Collection" (survey 2). And 10 further persons were mailed the questionnaire for judging similarities between persons on the basis of single free text interest phrases taken from the "Survey Collection".
From the 30 questionnaires mailed, 24 were returned (a surprisingly good average which is due to the fact that mostly close friends were asked to participate in the respective survey and that several "reminder" emails were sent to ensure the return of the questionnaires). Of the 25 returned sheets, 8 were answered for survey 1, 9 were answered for survey 2 and 7 answered were for survey 3.
The complete results of the survey plus statistical measures are shown in appendix C.
Original versions of the survey were designed to incorporate a substantial part of the test-collections and also incorporated control questions that were intended to measure the consistency

$\mathrm{sim}(\mathcal{X}^{(j_1)}, \mathcal{X}^{(j_2)})$(Part(I)

- Input:
  - Two sets of free text interest concepts $\mathcal{X}^{(j_1)}, \mathcal{X}^{(j_2)}$:

$$\begin{aligned} \mathcal{X}^{(j_1)} &= \{C_0^{(j_1)}, C_1^{(j_1)}, \ldots, C_{|\mathcal{X}^{(j_1)}|-1}^{(j_1)}\} \quad \text{and} \\ \mathcal{X}^{(j_2)} &= \{C_0^{(j_2)}, C_1^{(j_2)}, \ldots, C_{|\mathcal{X}^{(j_2)}|-1}^{(j_2)} \quad ; \end{aligned}$$

    where each of the concepts C has a free textual phrase representation of length $N(C)$:

$$\begin{aligned} C_i^{(j_1)} &\hat{=} \hat{C}_i^{(j_1)} = (w_{n_0}, w_{n_1}, \ldots, w_{n_{N(\hat{C}_i^{(j_1)})-1}}) \\ C_k^{(j_2)} &\hat{=} \hat{C}_k^{(j_2)} = (w_{m_0}, w_{m_1}, \ldots, w_{m_{N(\hat{C}_k^{(j_2)})-1}}) \end{aligned}$$

  - Required resources: WordNet 2.0; Microsoft Encarta Dictionary; Stop-list
- Preprocess:
  - Iterate over all free text interest phrases in both sets:

    for$(i, k = 0; i < |\mathcal{X}^{(j_1)}|, k < |\mathcal{X}^{(j_2)}|; i{+}{+}, k{+}{+})$
    Remove Stop-words (SR) and multiple instances of words (MIR):

$$\begin{aligned} \hat{C}_i^{(j_1)} \xrightarrow{\text{SR, MIR}} \widetilde{C}_i^{(j_1)} &= (w_{p_0}, w_{p_1}, \ldots, w_{p_{N(\widetilde{C}_i^{(j_1)})-1}}) \\ \hat{C}_k^{(j_2)} \xrightarrow{\text{SR, MIR}} \widetilde{C}_k^{(j_2)} &= (w_{q_0}, w_{q_1}, \ldots, w_{q_{N(\widetilde{C}_k^{(j_2)})-1}}) \end{aligned}$$

    endfor
- Prepare(I):
  - A Hashtable of all words $\mathscr{V}(\mathcal{X}^{(j_1)}) = \{w_{r_0}, w_{r_1}, \ldots w_{r_{|\mathscr{V}(j_1)|-1}}\}$ in $\mathcal{X}^{(j_1)}$ and a Hashtable of all words $\mathscr{V}(\mathcal{X}^{(j_2)}) = \{w_{s_0}, w_{s_1}, \ldots w_{s_{|\mathscr{V}(j_2)|-1}}\}$ in $\mathcal{X}^{(j_2)}$
  - Retrieve and preprocess Encarta Dictionary entries for all words:

    for$(a, b = 0\ ;\ a < |\mathscr{V}(\mathcal{X}^{(j_1)})|, b < |\mathscr{V}(\mathcal{X}^{(j_2)})|\ ;\ a{+}{+}, b{+}{+})$
    - Get dictionary entries for all senses for w_{p_a} and w_{q_b}. Concatenate them into $d(w_{p_a})$ and $d(w_{q_b})$.
    - Do stop-word removal, and compute word-vectors by counting term frequencies

$$\begin{aligned} d(w_{p_a}) \xrightarrow{SR} \widetilde{d}(w_{p_a}) \xrightarrow{CTF} \check{d}(w_{p_a}) &= (\nu_{u_0}, \nu_{u_1}, \ldots, \nu_{u_{|\mathscr{V}(\widetilde{d}(w_{p_a}))|-1}}) \\ d(w_{q_b}) \xrightarrow{SR} \widetilde{d}(w_{q_b}) \xrightarrow{CTF} \check{d}(w_{q_b}) &= (\nu_{l_0}, \nu_{l_1}, \ldots, \nu_{l_{|\mathscr{V}(\widetilde{d}(w_{q_b}))|-1}}) \end{aligned}$$

    endfor

Figure 4.9: Part I of the Algorithm for a Similarity Measure for Sets of Free Text Interest Phrases $\mathrm{sim}(\mathcal{X}^{(j_1)}, \mathcal{X}^{(j_2)})$

$\mathrm{sim}(\mathcal{X}^{(j_1)}, \mathcal{X}^{(j_2)})$ (Part(II)

- `Prepare(II):`
  - `Compute a similarity matrix` $\mathrm{Sim}^{\mathrm{dict}} = \mathscr{V}(\mathcal{X}^{(j_1)}) \times \mathscr{V}(\mathcal{X}^{(j_2)}) \rightarrow [0,1]$

$$\mathrm{Sim}^{\mathrm{dict}}_{ab} = \mathrm{sim}^{\mathrm{dict}}(w_{p_a}, w_{q_b}) =$$

$$= \begin{cases} 1 & \texttt{if} \quad w_{p_a} = w_{q_b} \\ \frac{1}{||\check{d}(w_{p_a})||||\check{d}(w_{q_b})||} \sum_{\{(e,f)|w_{u_e} \in \widetilde{d}(w_{p_a}) \wedge w_{l_f} \in \widetilde{d}(w_{q_b})\}} \nu_{u_e} \nu_{l_f} \; \delta_{u_e l_f} & \texttt{if} \quad w_{p_a}, w_{q_b} \in \texttt{Dictionary} \\ 0 & \texttt{else} \quad . \end{cases}$$

  - `Compute a similarity matrix` $\mathrm{Sim}^{\mathtt{WordNet}} = \mathscr{V}(\mathcal{X}^{(j_1)}) \times \mathscr{V}(\mathcal{X}^{(j_2)}) \rightarrow [0,1]$

$$\mathrm{Sim}^{\mathrm{WordNet}}_{ab} = \mathrm{sim}^{\mathrm{WordNet}}(w_{p_a}, w_{q_b}) =$$

$$= \begin{cases} 1 & \texttt{if} \quad w_{p_a} = w_{q_b} \\ \left(1 + \min_{\{(c,d)|w_{p_a} \in S_{t_c} \wedge w_{q_b} \in S_{t_d}\}} \left(\frac{pl(S_{t_c}, S_t^{\mathtt{ncsc}}) + pl(S_{t_d}, S_t^{\mathtt{ncsc}})}{2\, pl(S_t^{\mathtt{ncsc}}, S_0)^{\frac{1}{2}}} \right) \right)^{-1} & \texttt{if} \quad w_{p_a}, w_{q_b} \in \texttt{WordNet} \\ 0 & \texttt{else} \quad . \end{cases}$$

- `Compute` sim:
  - `Compare unordered sets of free text interest phrases:`

$$\mathtt{sim}(\mathcal{X}^{(j_1)}, \mathcal{X}^{(j_2)}) = \frac{1}{2} \frac{1}{|\mathcal{X}^{(j_1)}|} \sum_{i=0}^{|\mathcal{X}^{(j_1)}|-1} \max_k \mathtt{sim}(C_i^{(j_1)}, C_k^{(j_2)}) + \frac{1}{2} \frac{1}{|\mathcal{X}^{(j_2)}|} \sum_{k=0}^{|\mathcal{X}^{(j_2)}|-1} \max_i \mathtt{sim}(C_i^{(j_1)}, C_k^{(j_2)})$$

  - `where pairs of free text interest phrases are compared by:`

$$\begin{aligned} \mathtt{sim}(C_i^{(j_1)}, C_k^{(j_2)}) &= \mathtt{sim}(\widetilde{C}_i^{(j_1)}, \widetilde{C}_k^{(j_2)}) \\ &= \frac{1}{2} \frac{1}{N(\widetilde{C}_i^{(j_1)})} \sum_{a=0}^{N(\widetilde{C}_i^{(j_1)})-1} \max_b \mathtt{sim}(w_{p_a}, w_{q_b}) \\ &\quad + \frac{1}{2} \frac{1}{N(\widetilde{C}_k^{(j_2)})} \sum_{b=0}^{N(\widetilde{C}_k^{(j_2)})-1} \max_a \mathtt{sim}(w_{p_a}, w_{q_b}) \end{aligned}$$

  - `where pairs of words (lexicalized) concepts are compared by:`

$$\mathtt{sim}(w_{p_a}, w_{q_b}) = \frac{1}{\alpha + \beta} \left(\alpha \; \mathtt{sim}^{\mathtt{WordNet}}(w_{p_a}, w_{q_b}) + \beta \; \mathtt{sim}^{\mathtt{dict}}(w_{p_a}, w_{q_b}) \right)$$

- `Return` sim.

Figure 4.10: Part II of the Algorithm for a Similarity Measure for Sets of Free Text Interest Phrases $\mathrm{sim}(\mathcal{X}^{(j_1)}, \mathcal{X}^{(j_2)})$

of the ratings. What does that mean? Let the rating from user u for the semantic relatedness of items j_1 and j_2 be denoted by $r_u(j_1, j_2)$. The items of survey 1 are sets of free text interest phrases, the items of survey 2 are list-of-choice interest vectors and the items of survey 3 are single free text interest words and phrases. The numbering of the items is according to the numbers in the test collections "Survey Collection" (see figures C.1 and C.2 in the appendix) and "Dating Collection" (see figures C.3 and C.4 in the appendix) and the list of single free text interest words and phrases (see figure C.5 in the appendix).
In the original version of the surveys, users were asked to give a judgment $r_u(j_1, j_2)$ for items (j_1, j_2) and in other parts of the survey were asked to judge the semantic relatedness of the pair (j_2, j_1). Usually we would assume that $r_u(j_1, j_2) = r_u(j_2, j_1)$ since similarity measures are symmetric. Thus asking users for the relatedness of (j_1, j_2) **and** (j_2, j_1) would have allowed to study psycholinguistic effects of symmetry percipience and would also have established a mechanism to control consistency and thoroughness of the survey participants. The intention was to average the ratings for (j_1, j_2) and (j_2, j_1).
However, **test runs** with these extensive surveys (many elements from the test collections plus cross-checking with the lower triangle matrix of the similarity matrix) showed that over two hours were necessary to complete a single survey form which was far to long. Thus the decision was made to substantially reduce the extension of the survey and to abandon the cross checking.

4.7.7.1 Survey Evaluation

When **evaluating such a survey**, the first question that has to be answered is whether the ratings for a pair (j_1, j_2) of items from all participating users reflect a general tendency or whether the ratings are so subjective, that they appear to be **randomly distributed**. Judging whether a given set of ratings of n users $\{r_u(j_1, j_2) | 1 \leq u \leq n\}$ is randomly distributed or reflects a random distribution is usually accomplished with the help of **statistical tests**. In the present case, we could use a χ^2-Test to test (and reject) the hypothesis, that the ratings $\{r_u(j_1, j_2) | 1 \leq u \leq n\}$ are uniformly distributed for every pair (j_1, j_2). Another possibility is to use Gauß- or t-tests to check (and accept) the hypothesis that the ratings $\{r_u(j_1, j_2) | 1 \leq u \leq n\}$ are normal distributed $\{r_u(j_1, j_2) | 1 \leq u \leq n\} \sim \mathcal{N}(\mu_0, \sigma_0)$ for every pair (j_1, j_2).
Unfortunately, our sample sizes are far too small to make reasonable use of statistical tests. Therefore we must rely on conventional statistical measures such as expectation value and standard deviation. However, if sample sizes are extended (e.g. in future evaluations), statistical tests can be used to give quantitative evidence for the conclusions drawn here.

The first evaluation compares the overall rating behavior of the survey participants.

Table 4.2 shows the **average ratings** for every user together with the standard deviation of the user's ratings. What we can see from the table is that each user has a **bias** to generally judge either more strict or more generous as can be seen from the values for $\bar{r}_u$. In the first survey, user 2 has an average rating of only 2.06 while user 4 has an average rating of 4.67 which is substantially different. While survey 2 makes the users judge with a slightly more even bias, survey 3 seems more controversial with respect to bias. The most generous user had an average rating of 2.95 while the most strict user judged for $\bar{r}_u = 0.47$.
A possibility to rule out the differences in judging bias between the users would have been to normalize each judgement $r_u(j_1, j_2)$ to an average of 5 by setting $r'_u(j_1, j_2) = 5/\bar{r}_u \; r_u(j_1, j_2)$. For the sake of easy interpretability of the resulting statistical measures and for the sake of authenticity this was not done.

	Survey 1								
u	1	2	3	4	5	6	7	8	
$\bar{r}_u$	3.69	2.07	2.53	4.67	3.46	3.52	2.88	2.21	
σ_u	2.27	1.48	1.65	2.04	1.94	1.33	1.93	1.50	
	Survey 2								
u	1	2	3	4	5	6	7	8	9
$\bar{r}_u$	6.76	5.68	4.38	6.45	4.89	4.38	4.64	6.68	6.57
σ_u	2.01	1.40	2.37	2.09	1.87	1.84	2.04	1.82	1.26
	Survey 3								
u	1	2	3	4	5	6	7		
$\bar{r}_u$	2.31	1.53	2.06	0.67	0.47	2.95	1.71		
σ_u	2.26	2.16	2.94	1.90	1.87	2.76	2.84		

Table 4.2: The Average Ratings and Standard Deviations of Survey 1, 2 and 3 for all participants in the survey. The ratings are on a scale of 10 where 0 denotes np similarity and 10 denotes maximal similarity. σ_u denotes the Standard deviation estimated by ($\sigma_u = \sqrt{\frac{1}{99}\sum_{i,j}(r_u(j_1,j_2)-\bar{r}_u)^2}$ for surveys 1 and 2 and $\sigma_u = \sqrt{\frac{1}{399}\sum_{i,j}(r_u(j_1,j_2)-\bar{r}_u)^2}$ for survey 3). $\bar{r}_u$ denotes average rating for user u ($\bar{r}_u = \frac{1}{100}\sum_{i,j} r_u(j_1,j_2)$ for surveys 1 and 2 and $\bar{r}_u = \frac{1}{400}\sum_{i,j} r_u(j_1,j_2)$ for survey 3).

The **standard deviations** are comparable within all three surveys. Their numerical value shows that users do not tend to often give different extreme ratings but prefer to stay somewhat in the middle.
For a more detailed discussion see the sections below.

The most important goal of the survey was to measure the **quality of the similarity measures** for list-of-choice interest vectors and sets of free text interest phrases. Therefore we computed the average rating $Av(j_1,j_2) = \frac{1}{n}\sum_u r_u(j_1,j_2)$ for every pair of items (j_1,j_2) in the respective survey and computed the mean squared error between the average ratings $Av(j_1,j_2)$ and the similarity measure $\mathrm{sim}(j_1,j_2)$ for the respective surveys.
Assuming that the numbers $Av(j_1,j_2)$ and $\mathrm{sim}(j_1,j_2)*10.0$ were both stochastically independent and discreetly uniformly distributed ($X = Av(j_1,j_2) \sim \mathscr{U}(p)$ and $Y = \mathrm{sim}(j_1,j_2)*10.0 \sim \mathscr{U}(p)$) with probability $P[X=k] = P[Y=k] = \frac{1}{11}$ for $k \in [0,10]$ we would get for the square root of the mean squared error:

$$\begin{aligned} SQMSE(X,Y) &= \big(E((X-Y)^2)\big)^{\frac{1}{2}} \\ &= \Big(\sum_{xy}(x-y)^2\, f^{XY}(x,y)\Big)^{\frac{1}{2}} \\ &= \Big(\sum_{x=0}^{10}\sum_{y=0}^{10}(x-y)^2\,\big(\frac{1}{11}\big)^2\Big)^{\frac{1}{2}} \\ &= \Big(\big(\frac{1}{11}\big)^2\sum_{n=0}^{10}2n(11-n)^2\Big)^{\frac{1}{2}} \\ &= 4.472\ldots \quad . \end{aligned} \tag{4.17}$$

This number reflects the case where our similarity measure and the user ratings would have nothing to do with each other.

4.7.7.2 Results from Survey 1

For survey 1, the square root of the mean squared error between the average rating $Av(j_1, j_2)$ and the similarity measure for sets of free text interest phrases defined in figures 4.9 and 4.10 with $\alpha = 0.7$ (the weight for the Word-Net-based part of the sim-measure) and $\beta = 0.3$ (the part of the Encarta-based part of the sim-measure) yielded the result

$$SQMSE\Big(Av(j_1, j_2), (sim(x^{(j_1)}, x^{(j_2)}) * 10.0\Big) = 1.158 + O(10^{-4}).$$

This result shows that the correlation between the user ratings for the similarity of the free text interest phrases of survey 1 is excellent. The similarity measure performs **significantly better than random** (comparing with (4.17)). Furthermore, if we analyze the standard deviation of the votes of survey 1 shown in tables C.5 and C.6 in the appendix we get an average standard deviation of 1.62. This shows that (order of magnitude) the similarity measure performs well **within the boundaries of human judgement variations** which is the optimum that can be expected from a heuristic based similarity measure and is an **excellent result**.

We chose the parameters $\alpha = 0.7$ and $\beta = 0.3$ because we were not able to crawl all articles and dictionary entries from Encarta because no subscription was available. Thus a lot of similarities for words had to be set to zero in the corresponding part of the sim-measure. We thus gave the Encarta-based part of the sim-measure a lower weight.

Parameter Settings	$\alpha = 1.0$, $\beta = 0.0$	$\alpha = 0.7$, $\beta = 0.3$	$\alpha = 0.5$, $\beta = 0.5$	$\alpha = 0.3$, $\beta = 0.7$	$\alpha = 0.0$, $\beta = 1.0$
$SQMSE\Big(Av(j_1,j_2),(sim(x^{(j_1)},x^{(j_2)})*10.0\Big)$	1.73	1.15	1.36	1.65	2.14

Table 4.3: Varying the settings for α and β.

More systematically evaluating the parameter settings yields the results shown in table 4.3. We find that the influence of the second part of the similarity measure for words that uses the encyclopedic knowledge from Encarta is not beneficial in our experiment. This effect can be attributed to the fact that without a proper Encarta subscription we were only able to download about 50 per cent of the articles and dictionary entries for the single words. This introduces a distortion effect into the calculations. We conclude together with the findings from survey 3 (see below) that downloading all articles may deliver a result where we have a clearer minimum at a parameter setting of roughly $\alpha = 0.7$ and $beta = 0.3$.
It is nevertheless interesting that the lexical knowledge encoded in WordNet seems to be more appropriate for determining semantic relatedness than the encyclopedic knowledge from Encarta. We can also observe that even using the defective version of this second similarity measure part alone yields a correlation which is substantially better than random.

4.7.7.3 Results from Survey 2

For survey 2, the square root of the mean squared error between the average rating $Av(j_1, j_2)$ and the similarity measure for list-of-choice interest vectors defined in figure 4.5 using the upper taxonomy shown in figure 4.3 yielded the result

$$SQMSE\Big(Av(j_1, j_2), (sim(x^{(j_1)}, x^{(j_2)}) * 10.0\Big) = 1.958 + O(10^{-4}).$$

This result shows that the similarity measure performs **significantly better than random** (comparing with (4.17)). Furthermore, if we analyze the standard deviation of the votes of survey 2 shown in tables C.5 and C.6 in the appendix we get an average standard deviation of 1.66. This shows that (order of magnitude) the similarity measure performs **within the boundaries of human judgement variations** which is the optimum that can be expected from a heuristic based similarity measure and is an **excellent result**.

In terms of the **rating bias** among the participating users, it can be seen from table 4.2 that the average votes are higher than in case of survey 1. This is probably due to the fact that having to chose interests from a list (or taxonomy) of alternatives one the hand will trivially cause more identical phrase matches between two vectors. On the other hand it will make people choose the alternatives "less carefully", especially because there is no possibility to express a degree of interest in the respective interest alternatives and because the barrier to chose one or two more alternatives is lower than to state them explicitly (as in the case of free text phrases). Thus the vectors will generally appear more uniform than in case of free text phrases. The survey participants are thus faced with the problem of having to compare vectors which appear very similar. This explains why the votes are considerably higher than in case of survey 1.

4.7.7.4 Results from Survey 3

We will now examine the square root of the mean squared error between the average rating $Av(j_1, j_2)$ and the similarity Measure that compares single concept phrases from Sets of Free Text Interest Phrases. This similarity measure is defined in equation 4.12 and is the key part of the similarity measure for *Sets* of Free Text Interest Phrases defined in figures 4.9, 4.10. For the value of SQMSE we get the result

$$SQMSE\Big(Av(j_1, j_2), (sim(x^{(j_1)}, x^{(j_2)}) * 10.0\Big) = 2.191 + O(10^{-4}).$$

This result shows that the similarity measure also performs **significantly better than random** (comparing with (4.17)).
It performs slightly outside the bandwidth of judegment deviations of the survey participants which is characterized by an average standard deviation of 1.510. An obvious explanation are the missing Encarta entries which could not be downloaded. Another major influence factor is that WordNet also covers not the complete vocabulary that is reflected in the free text interest phrases which we used for the survey.
What is remarkable is that the measure for the single interest phrases performs worse than the overall similarity measure for complete sets of interest phrases. This a justification for the idea to chose the maximum similarity between a phrase from one set and all the phrases from the other set in the definition 4.5 instead of counting the similarity of all phrases compared with all phrases from the other set.

Parameter Settings	$\alpha = 1.0$, $\beta = 0.0$	$\alpha = 0.7$, $\beta = 0.3$	$\alpha = 0.5$, $\beta = 0.5$	$\alpha = 0.3$, $\beta = 0.7$	$\alpha = 0.0$, $\beta = 1.0$
$SQMSE\Big(Av(j_1,j_2),(sim(x^{(j_1)},x^{(j_2)})*10.0\Big)$	2.20	2.13	2.12	2.14	2.23

Table 4.4: Varying the settings for α and β.

Systematically evaluating the parameter settings yields the results shown in table 4.4. We find that the second part of the similarity measure for words that uses the encyclopedic knowledge from Encarta does not have a large influence in our experiment. We can observe a slight minimum of the $SQMSE$ measure at non-extreme settings for the parameters. This supports the conclusion that the Encarta-based second term which has the weight β has a positive effect on the similarity calculations.

4.7.8 Other Explict User Data

In the previous sections we have extensively discussed interest statements as an example for **explictly provided user data in a CIKS**. Besides interest statements, other explicit user data are contained in a CIKS that can be used for the detection of Ad-Hoc-Groups. One obvious example are **Buddylists**.
Buddylists are lists of user-names or other unique user-identifiers that can be freely and explicitly stated by every user of a community-service. These lists of friends are used to control access to personal data (as in case of the Studiosity community [55]) or to control communication. An example for the control of communication are Instant Messaging, where the buddylist is the central steering instrument to control in terms of whose instant messages reach one's terminal e.g.. Another example is the standalone mobile version of the jetzt.de community [35]. This porting of the mobile Studiosity platform is centered around the personal buddylist and made certain adaptations with respect to privacy. One example is that the FriendAlert service was reversed by informing a User A about the fact that a User B has entered A's proximity if B has A on his Buddylist.
Buddylists are explicit statements from users that document, which group they belong to. In chapter 6 we will further investigate, how buddylists can be used for Ad-Hoc-Group Information Management by combining them as crisp sets with other group models (e.g. fuzzy sets).

4.8 Communication Data with Tree-like Structure

In the previous section we have discussed explicit user data. We will now investigate implicitly won user data. In chapter 1 we have discussed that **communication** plays a tremendously important role in communities. We have modeled all actions in a community as communication acts mediated by a community's information- or knowledge space (CIKS). We have argued that text-based n:m semi-indirect communication is one of the most important forms of communication in a community. These forms of communication typically occur in CIKS-parts which have the form of **discussion boards or newsgroups** and their characteristics imply that a lot of social relationship information (including group structures) are implicitly encoded in these structures. We will therefore investigate these implicit structures and develop a similarity measure between persons communicating with the help of these structures with respect to Ad-Hoc-Group detection.
In order to properly define which structures are in the focus of this section the following criteria should be met:

- The structure should be a means for **text-based** communication.
- Users can **freely contribute** to the structure and can **freely access contributions** made by other users

- The contributions of users are preserved or **archived** [41] for a longer period of time.
- The structure is organized as a **tree** of **postings**:
  - A user makes his contributions in the form of postings. A posting is an **information item** in the sense of section 1.3.2 in chapter 1.
  - A posting is a **reply**, supplement, comment or other form of reaction (that we will summarize under the notion "reply") on an existing posting (usually written by another user).
  - A posting a has a **minimal set of meta-data**: The ID of the author and the ID of the posting b that the posting a is replying to.
  - The pairs (ID of a posting a, ID of the posting b that the posting a is replying to) form a **reply-relation**, which form the directed edges of a **tree graph**.

Examples for such communication structures are Usenet and discussion boards. **Usenet** is based on the NNTP protocol and is one of the oldest threaded distributed forms of text based communication on the internet. Each Newsgroup has a unique name and usually a distinct topic of discussion. Each top-level posting is root for a **thread**, a tree of replies and their respective follow up postings. **Web-based discussion boards** are a local adaptation of the idea of Usenet for single websites or community portals. Almost every larger website has a forum or discussion board attached to it and countless implementations for almost every data-base based web-technology (PHP, JSP etc.) exist.
Other forms of collaborative design of information with a strong communication character are **WIKIs** [130] (sometimes called CoWebs [110] (Collaborative WebPages)) (see chapter 1) which are also very interesting sources for implicit social- and group-information but will not be regarded here (see [110] for more).

4.8.1 Previous and Related Work

The literature on the measurement and interaction of social relations with computer supported communication is very vast. Two good books with many references are [31, 110]. Countless scientific projects exist in this field [41] which we will not attempt to review here. As an example, the paper [149] discusses the strength of social relations between two persons measured with the help of email conversation. The idea is that the relation is strong if e-mail between two persons is exchanged frequently, recently and reciprocally and is realized in a formula for the strength which is a function of user determined importance weights and the number of received and sent mails.
Even when the scope is limited to **tree-like communication structures** that adhere to the upper criteria, there are countless projects that try to collect implicit social relationship data from these structures.
In view of **visualizing social relationships** on that basis, the approaches [32, 164] are worth mentioning. The Loom system proposed in [32] aims at visualizing social relationships on the basis of Usenet conversation. The temporal development of the reply relation is visualized in a special format resembling a music score (see also [172]). The system also incorporates simple analysis of the posting content.
The approach in [164] uses several text analysis procedures to compute a net and visualization of social relations among authors, a set of discussion themes frequently used in the conversations

and a set of semantic networks that represent the main terms in the discussions and some relations among them. Various preprocessing steps analyze the messages and identify quotations, author signatures, reply relation and the index of authors. A part-of-speech tagger is used together with simple conversation tag analysis identifying relational words such as "if" "therefore", "consequently" etc. that give hints for logical relations between sentences or parts of sentences. This discourse analysis system is used "out of the box". The authors also use WordNet to compute a "lexical cohesion" index between a message and the messages that directly reply to or quote this message. The approaches for lexical cohesion (which we call semantic relatedness) that are used and cited in the paper are described in section 4.7.3.3 (see also [19]). The computed lexical ties are used to label the edges of the reply relation. The analysis results of the "lexical cohesion" of words resulting in a set of semantic nets of words used in the postings and the results of the conversation analysis are also used to adapt the visualization of the "discussion theme" network. The semantic nets of words are used to compute "themes" of discussion that connect the greatest number of authors.
Microsoft Netscan [139, 172] is another tool for searching, visualizing and analyzing newsgroups. It combines various visualization techniques and extensive analysis of the reply relation and some basic content analysis to allow for an improved navigation in the Usenet compared to conventional newsreaders.
Our approach introduced below does not specifically aim at improving these approaches with respect to binary relationship detection or categorization but is intended as a means of incorporating data mining on implicit communication data with tree like structure into the detection and analysis process of Ad-hoc-Groups.

4.8.2 Test Collections

For our experiments a news-crawler system was created [173] that allows to download all postings (content plus header) from one or more newsgroups. The crawler can store the postings either in a relational database or as separate files in the file-system. Furthermore, it is possible to filter out the quoted passages from other postings from the posting body.

4.8.3 Similarity Measure

As with all textual data, postings have to be extensively **preprocessed** into a numerical form (often called logical view) before they can be used as input patterns for Ad-Hoc-Group detection algorithms [161][6]. First step in the transformation is to **tokenize** the data-stream which represents the text into tokens (ordered, connected subsets of a text which contain no separation characters) which usually correspond to single words. After **Stop-Word-removal** (SR), the transformation often involves **stemming** in order to reduce the vocabulary. Stemming (S) is the linguistic operation of reducing word forms to their stem (e.g. reduce "swimming" and "swimmer" to "swim") by e.g. removing suffixes. For English the stemming algorithm of Porter is the most common method [6].
If independence assumptions are not made, noun-group-analysis can be incorporated which will be dropped here. The last and most important step in the transformation is the indexing and weighting step. This step can be followed by a feature selection step [51] which will be omitted here.

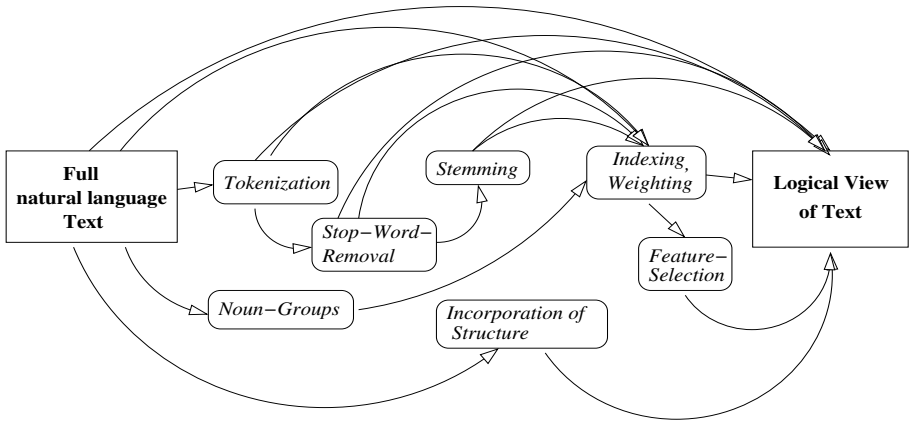


Figure 4.11: Transformation to Logical View[51]

4.8.3.1 Vector Model

Indexing (I) and **weighting** (W) (and to some extend also the other (pre)processing steps explained above) have as a prerequisite that a mathematical model is chosen which allows to capture as much of the semantics of the original text as possible (with respect to the application that one has in mind) while at the same generating a logical view of the text that can easily be handled by an algorithm. In most cases the characterization of a text within a model (the logical view) is generated by computing a numerical representation of a text. The most prominent model is the **vector model** which has been shortly mentioned in section 4.6.2. The vocabulary of all texts in a text-corpus determines a vector space where each element of the vocabulary corresponds to a dimension of this space. Each word in the vocabulary is weighted according to some weighting scheme. Thus a text document $d^{(j_1)}$ which is an ordered set of tokens (words) $\hat{d}^{(j_1)} = \{w_{p_0}, w_{p_1}, \ldots, w_{p_{|\hat{d}^{(j_1)}|}}\}$ is transferred into a vector of weights

$$\hat{d}^{(j_1)} \xrightarrow{\text{SR, S,...}} \widetilde{d}^{(j_1)} \xrightarrow{\text{I, W...}} \check{d}^{(j_1)} = (\check{w}_0, \check{w}_1, \ldots, \check{w}_{|\mathcal{V}|}). \tag{4.18}$$

where $\mathcal{V}$ denotes the vocabulary which is the set of all different non-stopword words (or word-stems) in all N documents in the collection $\{\hat{d}^{(j_1)} | 0 \leq j_1 \leq N-1\}$. Indexing delivers a global enumeration (dimension index) for the words (word-stems) in the vocabulary which allows us to uniquely associate a weight $\check{w}_q$ with a word (or word-stem) w_q of the vocabulary.
Thus, the semantics of a document is mainly represented as a direction in a high dimensional vector space. Other models represent a text with the help of probability distributions (e.g. Naive Bayes model).

In the vector model, the **weights** are in most cases computed as a function of the **term-frequency** $\nu_p^{(j_1)}$ that a word with index p occurs with in the document representation $\widetilde{d}^{(j_1)}$. The simple assumption is, that the more often a word occurs, the more significant it is for the semantics of $d^{(j_1)}$. Besides the term-frequency tf, **inverse document frequency** idf (equation 4.3) is often incorporated into the weights. As has been mentioned in section 4.6.2, this factor assumes that the more common a word is in all of the documents, the less expressive is its discriminative value as a criterion which allows to differentiate between the semantics of the single documents. In most cases, the weights are computed by multiplying tf and idf which

leads to the so called $tfidf$ weighting scheme:

$$\breve{w}_q^{(j_1)} = \nu_q^{(j_1)} \log \frac{N}{N_q} \tag{4.19}$$

In this form the model involves the **independence assumption**: the knowledge of weight $\breve{w}_q^{(j_1)}$ does not imply anything about a weight $\breve{w}_p^{(j_1)}$ with $p \neq q$. Thus we assume, that the weight of one word is independent of the weight of any other word. In real texts, words often occur in close proximity or even in pairs (**collocations**). E.g. in a text about computers the words "printer" and "driver" will be very likely to occur together which is of semantic significance. It has been shown that incorporation of knowledge about co-occurrence of terms does not necessarily improve performance for information retrieval tasks [6].

The most straightforward similarity measure between documents represented by vectors $\check{d}^{(j_1)}$ and $\check{d}^{(j_2)}$ is constructed by computing the cosine between these vectors with the inner product:

$$\mathrm{sim}^{\cos}(\check{d}^{(j_1)}, \check{d}^{(j_2)}) = \frac{1}{||\check{d}^{(j_1)}||||\check{d}^{(j_2)}||} \sum_{p=0}^{|\mathcal{V}|-1} \breve{w}_p^{(j_1)} \breve{w}_p^{(j_2)} \tag{4.20}$$

4.8.3.2 Comparing the Postings Contents

When constructing a similarity measure between two persons on the basis of their postings, the graph structure can be used as well as the contents of the postings. These two approaches can also be distinguished in the related work (see section 4.8.1). In contrast to web-pages which contain lots of formatting structures (HTML-Heading-Tags etc.) that can be analyzed and incorporated into the logical view (by e.g. simply counting words in HTML-headings multiple times to represent their amplified importance), postings do not contain such sub-structures. A straightforward incorporation the content of the postings of a user can be accomplished by representing the overall discussion interest of this user by a linear combination of the word vectors of all of his postings.

For the rest of the discussion we assume that a global enumeration index j for all postings in the tree-like discussion structure exists. If we denote the set of all m postings of user k with

$$\mathcal{K}_k = \{d_k^{(j_1)}, d_k^{(j_2)}, \ldots, d_k^{(j_m)}\}$$

the representation of the overall discussion interest x_k of this user is represented as

$$x_k = \frac{1}{m} \sum_{i=1}^{m} \check{d}_k^{(j_i)} \quad . \tag{4.21}$$

If we denote the reply relation by

$$\mathcal{R} = \{(j_{i_1}, j_{i_2}) \mid \text{ Posting } d^{(j_{i_1})} \text{ is a reply on posting } d^{(j_{i_2})} \} \tag{4.22}$$

and the reply relation constrained to postings from user k_1 replying to any posting from user k_2 by

$$\mathcal{R}^{(k_1)\to(k_2)} = \{(j_{i_1}, j_{i_1}) \mid \ (j_{i_1}, j_{i_2}) \in \mathcal{R} \ \wedge \ d^{(j_{i_1})} \in \mathcal{K}_{k_1} \wedge \ \wedge \ d^{(j_{i_2})} \in \mathcal{K}_{k_2} \ \} \tag{4.23}$$

we can also construct the linear combination of all replies of a user k_1 on any posting of user k_2

$$x_{(k_1)\to(k_2)} = \frac{1}{|\mathcal{R}^{(k_1)\to(k_2)}|} \sum_{\{i \mid \ \exists i_2 \, (j_i, j_{i_2}) \in \mathcal{R}^{(k_1)\to(k_2)}\}} \check{d}_{k_1}^{(j_i)} \tag{4.24}$$

4.8.3.3 Incorporation of Discussion Thread Structure

In general we can assume, as has been mentioned before, that a discussion board is represented at any given point in time by a tree of postings. The root node represents the overall topic of the discussion board. A posting has a child posting if the child posting is a reply to the father posting. The reply relation can (for the moment) be loosely defined as "dealing with the same (sub)-topic". A user can chose, whether he wants to position his contribution either as an independent posting (placing it as a child of the root node) or as a child of some posting. This ensures that on average the answer relation between two postings is not a deliberate outcome of some statistical classification algorithm but rather user-intended.
From the fact that every posting has exactly one author we can compute various graphs. Figure 4.12 depicts one possible graph where the authors of the postings have a relation to one another that is characterized by the number of replies to a posting from one of the authors. Another

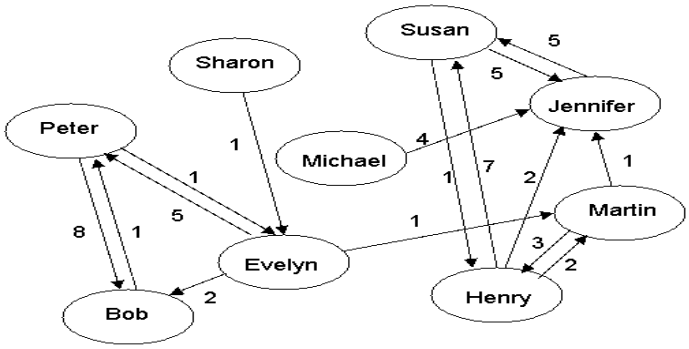


Figure 4.12: Graph computed from the reply relation. The numbers indicate the number of postings that author k_1 has written in reply to one posting from author k_2 (The origin of the arcs corresponds to author k_1)

possibility could be to directly work on the graph of the reply relation.
There are various possibilities to compute groups of people from such graphs. Some possibilities are indicated in [188]. Another possibility is to transform the frequency of the interaction in form of replies into a similarity measure between two people which can then be used in an Ad-Hoc-group detection algorithm.
Regarding this similarity measure (which can be interpreted as a measure of intensity of the social ties between two persons), two observations can be made:

(1) The larger the difference between the numbers of replies is, the lower is the intensity of interaction. E.g. if person k_1 replied only once to a posting of person k_2, but person k_2 replied 14 times to postings from person k_1, it can be concluded that person k_2 has a substantial interest in the postings of person k_1 but person k_1 does not seem to have much interest in the postings of user k_2. Thus the mutual interest is low.

(2) The larger the number of mutual replies is, the more intense is the interaction of the two persons.

Thus, a possible similarity measure between person k_1 and person k_2 could be constructed like this (denoting the number of replies from person k_1 to postings authored by person k_2 by $m_{k_1 \to k_2}$ and denoting the number of replies from person k_2 to postings authored by person k_1 by $m_{k_2 \to k_1}$ respectively).

$$\mathrm{sim}(k_1, k_2) = 1 - \frac{1}{(1 + m_{k_1 \to k_2} m_{k_2 \to k_1} \exp(-\frac{|m_{k_1 \to k_2} - m_{k_2 \to k_1}|^2}{\sigma^2}))^q} \tag{4.25}$$

The factor $m_{k_1 \to k_2} m_{k_2 \to k_1}$ in the denominator was introduced in accordance with observation (2) above and the factor $\exp(-\frac{|m_{k_1 \to k_2} - m_{k_2 \to k_1}|^2}{\sigma^2})$ was introduced in accordance with observation (1) above. The exponent q and the deviation σ are parameters which can be adapted.

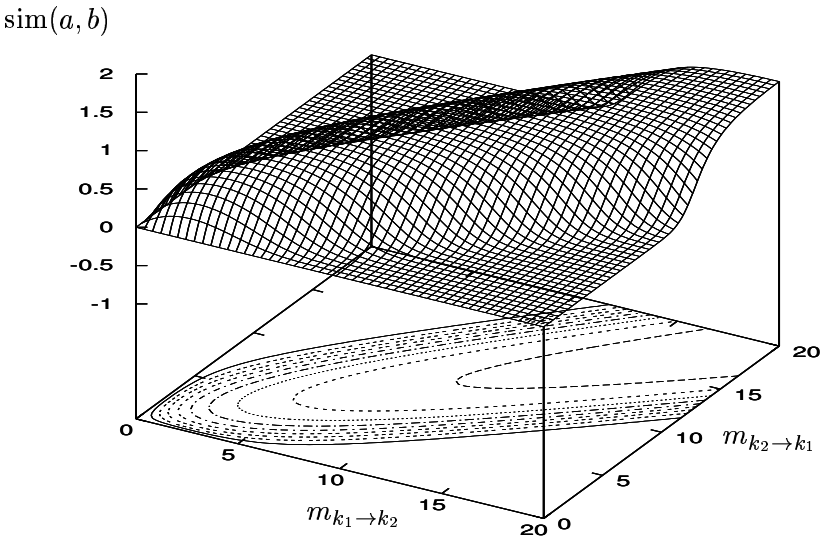


Figure 4.13: The function $1 - 1/(1 + m_{k_1 \to k_2} m_{k_2 \to k_1} \exp(-\frac{|m_{k_1 \to k_2} - m_{k_2 \to k_1}|^2}{\sigma^2}))^q$ with $\sigma^2 = 10$ and $q = 0.5$. Contour lines range from 0.1 to 0.9.

Of course, the measure of the intensity of social interaction cannot qualify whether the interaction is of positive nature or of negative nature. This becomes especially apparent in so called "flame wars" which are discussion threads where people with opposing opinions discuss a certain matter with high intensity.
Nevertheless it is fair to assume, that two persons with a high interaction intensity are interested in the same topics which are discussed in the course of the interaction.
Furthermore, the graph in figure 4.13 can be augmented by adding the linear combination of the word-vectors of the reply-postings as additional "weights" to the edges. This yields the "topic" of the interaction. Thus the directed graph edge between person k_1 and person k_2 would be tagged with the number $m_{k_1 \to k_2}$ and the linear combination of the word vectors of the replies $x_{(k_1) \to (k_2)}$.

Extensions In order to enlarge the expressiveness of the distilled entities $m_{k_1 \to k_2}$ and $x_{(k_1) \to (k_2)}$, we can incorporate more structural elements of the posting tree. Instead of only regarding immediate replies we can also regard **weighted indirect replies**. If a top-level-posting is **root of a discussion tree**, it is assumed that the posting **defines a general topic** or thematic

direction for the postings in the tree below it. This assumption can be continued to the nodes within the tree as well. The assumption corresponds to a general convention that governs the use of tree-like communication media such as discussion boards.
If it is further assumed that **postings become more specific** as the **levels of the tree increase**. We can then incorporate these two assumptions into the computation of $m_{k_1 \to k_2}$ and $x_{(k_1)\to(k_2)}$ by regarding the indirect answers to postings on a higher tree level.
For this purpose we will have to broaden our definitions from section 4.8.3.2. We will define a reply relation of degree n. To do that, we denote the level of a posting d in a posting tree by $l(d)$ and the set of children of a posting d in a posting tree by $\Omega(d)$. Thus the reply relation of degree n will be defined as

$$\mathcal{R}_n = \{(j_{i_1}, j_{i_2}) \mid \text{ Posting } d^{(j_{i_1})} \in \Omega(d^{(j_{i_2})}) \ \wedge \ l(d^{(j_{i_2})}) - l(d^{(j_{i_1})}) = n \ \} \tag{4.26}$$

which will make $\mathcal{R} = \mathcal{R}_1$ (compare definition (4.22)). The reply relation constrained to postings from user k_1 replying to any posting from user k_2 of degree n is then defined as

$$\mathcal{R}_n^{(k_1)\to(k_2)} = \{(j_{i_1}, j_{i_2}) \mid \ (j_{i_1}, j_{i_2}) \in \mathcal{R}_n \ \wedge \ d^{(j_{i_1})} \in \mathcal{K}_{k_1} \wedge \ \wedge d^{(j_{i_2})} \in \mathcal{K}_{k_2} \ \} \tag{4.27}$$

we can then define a weighted version of the quantity $m_{k_1 \to k_2}$ by setting

$$m_{k_1 \to k_2} = \frac{1}{\sum_{n=1}^{l_{\max}} w_n} \sum_{n=1}^{l_{\max}} w_n \, |\mathcal{R}_n^{(k_1)\to(k_2)}| \tag{4.28}$$

which also counts the indirect replies. The weights should be chosen so that $w_n < w_{n+1}$. For example we could set $w_n = 2^{n-1}$.
The linear combination of all replies of a user k_1 on any posting of user k_2 would then have to be defined as

$$x_{(k_1)\to(k_2)} = \frac{1}{\sum_{n=1}^{l_{\max}} w_n} \sum_{n=1}^{l_{\max}} w_n \, \frac{1}{|\mathcal{R}_n^{(k_1)\to(k_2)}|} \sum_{\{i \mid \ \exists i_2 \, (j_i, j_{i_2}) \in \mathcal{R}_n^{(k_1)\to(k_2)}\}} \check{d}_{k_1}^{(j_i)} \tag{4.29}$$

4.8.3.4 Combining Structure and Content

If we combine equation (4.25) and the cosine similarity measure (4.20) for the comparison of the word vectors

$$\begin{aligned} x_{(k_1)\to(k_2)} &= (\breve{w}_0^{(x_{(k_1)\to(k_2)})}, \breve{w}_1^{x_{(k_1)\to(k_2)}}, \ldots \breve{w}_{|\mathcal{V}|-1}^{(x_{(k_1)\to(k_2)})}) \\ \text{and } x_{(k_2)\to(k_1)} &= (\breve{w}_0^{(x_{(k_2)\to(k_1)})}, \breve{w}_1^{(x_{(k_2)\to(k_1)})}, \ldots \breve{w}_{|\mathcal{V}|-1}^{(x_{(k_2)\to(k_1)})}) \end{aligned}$$

an overall similarity measure can be constructed:

$$\begin{aligned} \text{sim}(a,b) = \frac{1}{\alpha+\beta} \Bigg(& \alpha \ \Big(1 - \frac{1}{(1 + m_{k_1\to k_2} m_{k_2 \to k_1} \exp(-\frac{|m_{k_1\to k_2} - m_{k_2\to k_1}|^2}{\sigma^2}))^q}\Big) \\ & + \ \beta \ \Big(\frac{1}{||x_{(k_1)\to(k_2)}|| \, ||x_{(k_2)\to(k_1)}||} \sum_{p=0}^{|\mathcal{V}|-1} \breve{w}_p^{(x_{(k_1)\to(k_2)})} \breve{w}_p^{(x_{(k_2)\to(k_1)})}\Big)\Bigg) \end{aligned} \tag{4.30}$$

We will use this measure for the calculations in the next chapter.
A survey for checking this similarity measure seems to be too complicated to conduct, because the participants would have to read a large amount of (large) postings in order to deliver sufficient data that could be compared to the measure. This would be beyond the cognitive scope of the participants at least if a reproducible and meaningful is expected. A testing scenario for such a measure would have to be indirect by testing the acceptance of the application that uses the measure.

Summary

In the first part of this chapter, two similarity measures are the main outcome. The first measure allows to compare two sets of free text interest phrases and the second measure allows to compare two vectors of list-of-choice interests. To be able do this, we first extensively discussed what principle types of interest statements exist in communities on the internet and discussed the various NLP difficulties that arise when preprocessing and comparing these interests with one another. In case of the List-of-Choice Interest Vectors, a thorough discussion on topic taxonomies was necessary to allow for a semantically rich similarity measure to be developed. In case of the Free Text Interest Phrases, we first had to develop means to semantically compare words with the help of lexical sematic ressources such as WordNet. We then had to combine the semantic distances between words to arrive at the level of individual phrases and then had to perform another step to arrive at a measure to compare sets of such phrases in a meaningful way. Several surveys were conducted to measure the subjective similarity judgements of humans when confronted with interest data. The surveys were quantitatively analyzed and compared with the numerical outcome of the proposed similarity measures and an excellent congruence was achieved. Finally we throughly analyzed communication data with tree-like structure and combined state of the art vector model based content analysis and similarity with similarity based on the reply relations reflected in the tree structure of the communication data. All the developed similarity measures allow for a good comparison between individual users in view of the detection and modeling of groups which is discussed in the next chapters.

Chapter 5

Group Detection and Modeling Algorithms

This chapter is devoted to the development of methods for the detection and modeling of Ad-Hoc-Groups and abstract groups. The first part deals with the detection and the modeling of Ad-Hoc-Groups and associated abstract groups on the basis of location and velocity. We start by shortly reviewing the most important notions and notations from crisp clustering and introduce the most common algorithms and crisp cluster validation approaches. We then describe the conduction of basic experiments with crisp clustering on the SUMI simulation data and develop a method for socially motivated cluster validation and selection. In order to be able to compare the group data from the simulation with the group data from the group detection algorithms, we then throughly develop quantitative mathematical measures. On the basis of these measures and the knowledge from SUMI simulation we then systematically investigate the influence of the choice of the clustering algorithms, the influence of the choice of distance measure and conventional crisp clustering methods and the influence of varying the characteristic parameters of socially motivated cluster selection on the group detection and modeling algorithms. The follwing part is devoted to the detection and modeling of the underlying abstract groups that the found Ad-Hoc-Groups are instantiations of. The members structure of the groups and the periodicity of their occurence are taken as criteria for the abstract group detection method whose performance is finally also quantitatively investigated. The second part of the chapter deals with detection and modeling of abstract groups on the basis of interests and communication data. These type of data require fuzzy clustering approaches which are shortly reviewed together with fuzzy cluster validation strategies. Since our approach is based on the similarity measures developed in chapter 4, we investigate relational fuzzy clustering approaches as a the main tool. Quantitative experiments are decribed and improvements of the algorithms in the literature are developed. We describe the qualitative behaviour of the developed group detection and modeling procedures on the collected test data. The chapter is concluded by discussing methods for the combination of group models.

5.1 Detection and Modeling of Ad-Hoc-Groups and Abstract Groups on the Basis of Spatio-Temporal Proximity and Velocity

As was discussed in chapters 2 and 3 spatio-temporal proximity with respect to **location and velocity** may be considered a **good indicator** for an **Ad-Hoc-Group**. In this section we will therefore investigate and further develop techniques which identify such socially relevant instantiations of existing **abstract groups** and the underlying abstract groups themselves on the basis of time, space and velocity (that is in a Location Phase Space). According to the program of section 2.4, the main toolset that we will operate with are **clustering algorithms**. We will therefore first supplement the brief introduction into crisp clustering algorithms of section 2.1.2.3 by taking a slightly more general point of view in order to clarify background, notions and notation.

5.1.1 Clustering

In the language of data mining, groups correspond to clusters of users. Clustering methods are key methods of data mining that have the goal to **partition** a given set of **data into clusters** [77] [161]. Clustering methods are **unsupervised classification** methods. That means that the construction of the classifier is not guided by providing the algorithm with a training set of examples with pre-assigned class-labels. The construction of the classifier / clusterer is performed on the basis of the raw pattern data alone and the clusterer has to detect class structures based on intrinsic attributes of the pattern set.

In most cases clustering algorithms are used only as a means for data analysis without the goal of using the results after the learning process as an actual classifier for classifying new, previously unseen patterns.

Thus, a clustering algorithm is generally viewed as an unsupervised learning algorithm which learns a **partition** $\mathscr{C} : \mathcal{X} \to \mathcal{I}$ of a set (a universe) $\mathcal{X}$ where $\mathcal{I}$ is a set of cluster/class-indices. As has been mentioned before, the partition can be classified according to various aspects which are not "orthogonal" (e.g. a fuzzy partition is never exclusive). The major classification dimensions are [77]:

- **Exclusive vs. Non-Exclusive**
- **Crisp vs. Fuzzy**
- **Hierarchical vs. Non-Hierarchical**

Exclusive Partitions assign every object to exactly one cluster, whereas **non-exclusive** partitions allow for an object to be member of several clusters (overlapping clusters).

Hierarchical partitions impose a tree structure on the clusters where an edge $\mathcal{C}_j \to \mathcal{C}_i$ implies $\mathcal{C}_j \subset \mathcal{C}_i$. Visualizations of these trees are called **Dendrograms**. In general we have $\mathcal{C}_1 = \mathcal{X}$ as root node and one-element classes $\mathcal{C}_i = \{x_k\}$ as leafs. [1] **Non-Hierarchical** cluster structures do not have this property.

[1] Notation (throughout this chapter): vectors are denoted as x_k (or π_i etc.). The components of these vectors are denoted as $(x_k)_j$ (or $(\pi_i)_l$ etc.).

Crisp partitions $\mathscr{C}^{crisp}$ assume conventional characteristic functions μ_i for the clusters $\mathcal{C}_i \subseteq \mathcal{X}$. This means that either a pattern is a member of a given cluster or it is not.

$$\mu_i : \mathcal{X} \to \{0,1\} \text{ with } \mu_i(x \in \mathcal{X}) = \begin{cases} 1 & x \in \mathcal{C}_i \\ 0 & x \notin \mathcal{C}_i \end{cases} \tag{5.1}$$

(Thus we have $\mathscr{C}^{crisp}(x) = i \leftrightarrow \mu_i(x) = 1$).

Fuzzy partitions assume fuzzy clusters. A fuzzy set $\mathcal{C}_i$ is characterized by its fuzzy membership function μ_i:

$$\mu_i : \mathcal{X} \to [0,1] \tag{5.2}$$

which assigns to an object in $\mathcal{X}$ a continuous degree of membership in the fuzzy set. Fuzzy partitions are a special form of non-exclusive partitions.
A fuzzy classifier / clusterer assigns to a pattern a degree of membership in every cluster. The result is a fuzzy partition (clustering, classification) of the set of patterns $\mathcal{X}$.

Since in most cases we a dealing with discrete countable sets $\mathcal{X}$ we can view the values of the membership-functions $\mu_i(x_k \in \mathcal{X})$ as a matrix $\mu_i(x_k \in \mathcal{X}) = U_{ik}$.

The actual **algorithms** that perform the clustering are manifold and its hard to distill a consistent view on the numerous algorithms because there are so many contributions from different scientific areas. However, clustering algorithms can be classified according to some basic properties, the most important of which are [77]

- **Agglomerative vs. divisive**
- **Serial vs. simultaneous**
- **Monothetic vs. polythetic**
- **Graph-theoretic vs. Matrix-Algebra**
- **Representation of input data**

Agglomerative algorithms start with one cluster for each object and in the course of the algorithm merge clusters until all clusters have been merged into the original set $\mathcal{X}$. **Divisive** algorithms start with $\mathcal{X}$ and perform subdivisions until the partition with one object per cluster is reached. Serial algorithms consider the objects one by one, simultaneous algorithms consider all objects at once.

As has been pointed out before, we generally assume an object x to be represented by an attribute-value- or feature vector. If $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$ and we have m attributes (the feature space is m-dimensional) we can represent $\mathcal{X}$ as a $n \times m$ feature matrix X_{ij}. **Montothetic** algorithms use the features one by one, whereas **polythetic** algorithms use them all at once. Algorithms which operate on matrix representations like the feature matrix usually have a **Linear Algebra view** whereas other algorithms use graphs and **graph-theoretic approaches** to describe the objetcs, their relations and the clustering operations. However, in many cases these views can be transformed into one another and thus do not represent a distinctive feature of clustering algorithms.

Another very interesting point is the form and representation of the **input data** that the algorithm operates on. Many algorithms require the specification of the desired number of clusters. In general, algorithms take as further input either the pattern matrix X_{ij} or the Proximity matrix. The **Proximity matrix** (sometimes also called Relational Matrix) is a symmetric $n \times n$ matrix $R_{k_1 k_2}$ that specifies for every pair of objects $(x_{k_1}, x_{k_2}) \in \mathcal{X}^2$ a proximity value $\text{prox}(x_{k_1}, x_{k_2})$. A proximity value can be a similarity or dissimilarity (e.g. distance) value for the two objects. The **pattern matrix** can contain **binary, discrete or continuous numbers**. Both matrices can be populated with values whith a **nominal** significance or just an **ordinal** significance etc. The possibilities are manifold.
Furthermore, a vast amount of **data-preprocessing techniques** exists: Information-theoretic feature selection methods, principal component analysis, dimension reductions etc.. These techniques aim at revealing the "true" representation of the data. Obviously, since there is no universal truth or notion of "best" "true" etc., the algorithms for preprocessing need to be chosen adapted to the current domain [161].

5.1.2 Crisp Clustering

For the detection of Ad-Hoc-Groups with respect to location and velocities, several qualitative experiments with various clustering algorithms were conducted. In these experiments, fuzzy clustering algorithms seemed less suitable for detecting Ad-Hoc-Groups than crisp clustering techniques. We will briefly describe the results of fuzzy clustering when we introduce these class of algorithms in more depth later in this chapter.
We will now describe the three most common crisp clustering algorithms that we have used for our experiments in the course of the development of a reliable crisp clustering based detection method for Ad-Hoc-Groups with respect to location and velocities. The choice for these algorithms was made on the basis of their commonness (determined by studying standard literature) and availability of their implementations (developed in the course of [37]))

5.1.2.1 SAHN

SAHN clustering was already introduced in section 2.1.2.3. As was stated there the idea of this sequential agglomerative hierarchical non-overlapping family of algorithms is to start from a partition where very element of $\mathcal{X}$ is assigned its own cluster and subsequently merging these clusters until terminating at a single cluster containing all the elements of $\mathcal{X}$. In that way a cluster hierarchy is computed which is conveniently displayed with the help of a **Dendrogram**. The many variants of this family are mostly defined by the measure which is used to compare two clusters in the intermediate steps of the algorithm. Usually the problem is formulated with the help of a distance measure $d(\mathcal{C}_{i_1}, \mathcal{C}_{i_2})$. In each step, the two clusters with the minimal distance are merged. The most common variants for d are [161] **complete link** where

$$d(\mathcal{C}_{i_1}, \mathcal{C}_{i_2}) = \max_{\{k_1,k_2 | x_{k_1} \in \mathcal{C}_{i_1} \wedge x_{k_2} \in \mathcal{C}_{i_2}\}} ||x_{k_1} - x_{k_2}|| \tag{5.3}$$

and **single link** where

$$d(\mathcal{C}_{i_1}, \mathcal{C}_{i_2}) = \min_{\{k_1,k_2 | x_{k_1} \in \mathcal{C}_{i_1} \wedge x_{k_2} \in \mathcal{C}_{i_2}\}} ||x_{k_1} - x_{k_2}||. \tag{5.4}$$

This is in complete correspondence to the graph theoretic formulation given in figure 2.3. Note that in the given formulations, SAHN is only practical on metric pattern spaces.

5.1.2.2 K-Means-Clustering

K-Means Clustering technique follows a slightly different paradigm in that clusters $\mathcal{C}_i$ are described by **prototypes** π_i. In order to assign patterns x_k to clusters, the well known **nearest neighbor rule** is applied:

$$\mathscr{C}(x_k) = i_a \leftrightarrow ||x_k - \pi_{i_a}|| = \min_i ||x_k - \pi_i|| \tag{5.5}$$

that is the pattern is assigned to cluster $\mathcal{C}_{i_a}$ if its distance to the cluster's prototype is minimal. This nearest neighbour paradigm is well known in other branches of machine learning (e.g. text-classification [1]).
In order to determine the prototypes, an **optimization technique** can be used which gives the method its name. Formulating the clustering problem as an optimization problem is often guided by the paradigm of minimizing inter-cluster cohesion and maximizing intra-cluster cohesion. This can be achieved by minimizing the quadratic distances between the patterns and the prototypes, that is minimizing the square error

$$J_{\text{SQE}} = \sum_{i=1}^{c=|\mathcal{I}|} \sum_{\{k|x_k \in \mathcal{C}_i\}} ||x_k - \pi_i||^2 \tag{5.6}$$

From the necessary condition for a local extremum $\frac{\mathrm{d}J_{\text{SQE}}}{\mathrm{d}\pi_i} = 0$ we get an expression for the cluster prototypes

$$\pi^i = \frac{1}{|\mathcal{C}_i|} \sum_{\{k|x_k \in \mathcal{C}_i\}} x_k \tag{5.7}$$

that is the cluster prototypes (cluster centers) are given by the "center of gravity" of its associated patterns.
Since the association of the patterns to the clusters (via the nearest neighbor role) depends on the cluster centers and since the computation of the cluster centers depends on the association of the patterns to the clusters, the k-means algorithm uses **alternating optimization** [161] which randomly initializes the set Π of prototypes and iterates by computing the association of the patterns to the clusters and the set Π of prototypes. The stopping criterion can be determined by a bound to the changes in the set Π.

5.1.2.3 Minimum Spanning Tree Clustering

According to [78], the minimum spanning tree algorithm is the best known graph-theoretic divisive clustering algorithm. Since it does not use prototypes, it can be used in cases where only a matrix of distances between the patterns is known. This matrix, of course, corresponds to a weighted graph. The algorithm first constructs a minimum spanning tree for this weighted graph. It then subsequently removes the edges with the highest weight (corresponding to the largest distance) until the desired number of clusters is achieved or until a complete dendrogram is computed.

K-Means Clustering

- Input:
  - Number of clusters c.
  - Set of patterns $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$
  - Maximum number of iterations $t_{\max}$
  - stopping criterion bound ϵ
  - Suitable metric $||\ ||$
- Initialize:
  - Determine randomly the initial set of prototypes $\Pi^{(0)} = \{\pi_1{}^{(0)}, \pi_2{}^{(0)}, \ldots, \pi_c{}^{(0)}\}$
- Process:
  - For (step $t := 0$; $t \leq t_{\max}$; t++)
    - Compute the association of the patterns to the clusters with the nearest neighbor rule (Equation 5.5)
    - Compute the prototypes $\Pi^{(t)}$ with Equation 5.7
    - If ($\forall i\ ||\pi_i{}^{(t)} - \pi_i{}^{(t-1)}|| < \epsilon$) break.

    Endfor

Figure 5.1: Algorithm for K-Means Clustering (adapted from [161])

5.1.2.4 Crisp Cluster Validation Strategies

All three of the algorithms introduced before require to state the **number of clusters on input**. The graph theoretic algorithms require this input parameter to determine the cutting level of the dendrogram to actually produce a partition.
Since this quantity is usually not known in advance, it is the usual practice to run the algorithms with various number of cluster inputs and subsequently compare the resulting clusterings. **cluster validation** strategies provide criteria for this comparison process.
The application of clustering persons from location and speed data implies that no a-priori knowledge about the number of clusters exists. Furthermore, the algorithms try to find clusters no matter if a cluster structure that is meaningful for the desired applications exists.

In order to find the right setting for the parameters of the clustering algorithms that are used, and especially to find the right number of clusters c, literature proposes countless methods [77][60][61]. One possibility to classify the criteria is to divide them into external, internal and relative criteria.
External citeria measure the quality of a clustering by comparing it to a manually pre-assigned set of class indices, like it is the case in supervised learning (e.g. text-classification methods).

Since in our application scenario we do not have any pre-assigned classifications, external criteria cannot be used. **Relative criteria** compare the results of different clustering schemes or the results for different parameter settings of one algorithm with one another in order to optimize the parameter settings or find the right number of clusters. **Internal criteria** evaluate clustering results based on the input patterns themselves and the clustering results.
Other aspects of such validation measures is whether they are statistically motivated or optimizationally motivated. **Statistically motivated** validation makes assumptions about the randomness of the clustering and attempts to prove the validity of a clustering with the help of statistical tests. **Optimizationally motivated** techniques propose a numerical function which measures the "quality" of the clustering. A simple optimizationally motivated quality criterion is the mean **square error criterion** of equation 5.6. The goal of all statistical measures for cluster validation is to weigh inter-cluster coherence (or compactness) and intra cluster coherence. The mean square error criterion only measures the inter cluster compactness. The smaller J_{SQE} is, the greater is the compactness of the computed clusters and the better the clustering. In order to find e.g. the right number of clusters for the c-means algorithm, the algorithm is run with different values for c. We then have to find a local minimum for J_{SQE}. Local minimum because the global minimum for J_{SQE} is always reached by placing every pattern x_k in its own cluster ($c = n$), which results in $J_{SQE} = 0$.
Another well known general cluster validation criterion is the **Dunn Index** [60]

$$D = \min_{i_1 \in [1,c]} \left(\min_{i_2 \in [1,c]} \left(\frac{d_1(\mathcal{C}_{i_1}, \mathcal{C}_{i_2})}{\max_{i_3 \in [1,c]} d_2(\mathcal{C}_{i_3})} \right) \right) \tag{5.8}$$

where $d_1(\mathcal{C}_{i_1}, \mathcal{C}_{i_2})$ is the distance function between two clusters defined by

$$d_1(\mathcal{C}_{i_1}, \mathcal{C}_{i_2}) = \min_{\{(k_1,k_2) | x_{k_1} \in \mathcal{C}_{i_1} \wedge x_{k_2} \in \mathcal{C}_{i_2}\}} ||x_{k_1} - x_{k_2}|| \tag{5.9}$$

(that is the single link distance from SAHN).
The "diameter" d_2 of the clusters is defined by

$$d_2(\mathcal{C}_i) = \max_{\{(k_1,k_2) | x_{k_1} \in \mathcal{C}_i \wedge x_{k_2} \in \mathcal{C}_i\}} ||x_{k_1} - x_{k_2}|| \tag{5.10}$$

The pragmatics behind this index is again motivated by the demand for inter cluster compactness and intra cluster separation: If clusters are compact and well separated the distance between the clusters is large and their diameter is small.

5.1.3 Basic Experiments

In this section we will describe the basic clustering-based group detection algorithm and present several quantitative experiments which test the quality of the technique and allow for determining key parameters of the method.

5.1.3.1 The Role of SUMI

As has been extensively discussed in chapter 3, real location and velocity data are impossible to collect at the moment with reasonable accuracy. Therefore the **SUMI simulation toolkit** was developed in order to produce realistic position and movement data. Through the special approach of seamlessly combining individual motion with group motion, it is possible to access all relevant group data of the simulation at any time, allowing to state for every iteration step,

- **which groups are active** in this step (the Ad-hoc-groups),
- **which members** of a group are "on their way" to the meeting point and which have already reached it.
- wheter it is a **random group** (corresponding to an Ad-Hoc-Group whose abstract existence is limited to this generalized spatio-temporal situation)
- or wheter it is a **regular group** (corresponding to an Ad-Hoc-Group whose corresponding abstract group is instantiated in further generalized spatio temporal situations in the form of further Ad-Hoc-Groups)

This **knowledge** is necessary to **quantitatively evaluate** the group detection algorithms. Even for highly accurate real location data, this knowledge would not be easily available (which is the motivation for the development of an automatic detection technique) and thus we would have to rely on complicated and resource consuming surveys to evaluate the quality of our algorithms. This is the second and even more important motivation to use simulated data.

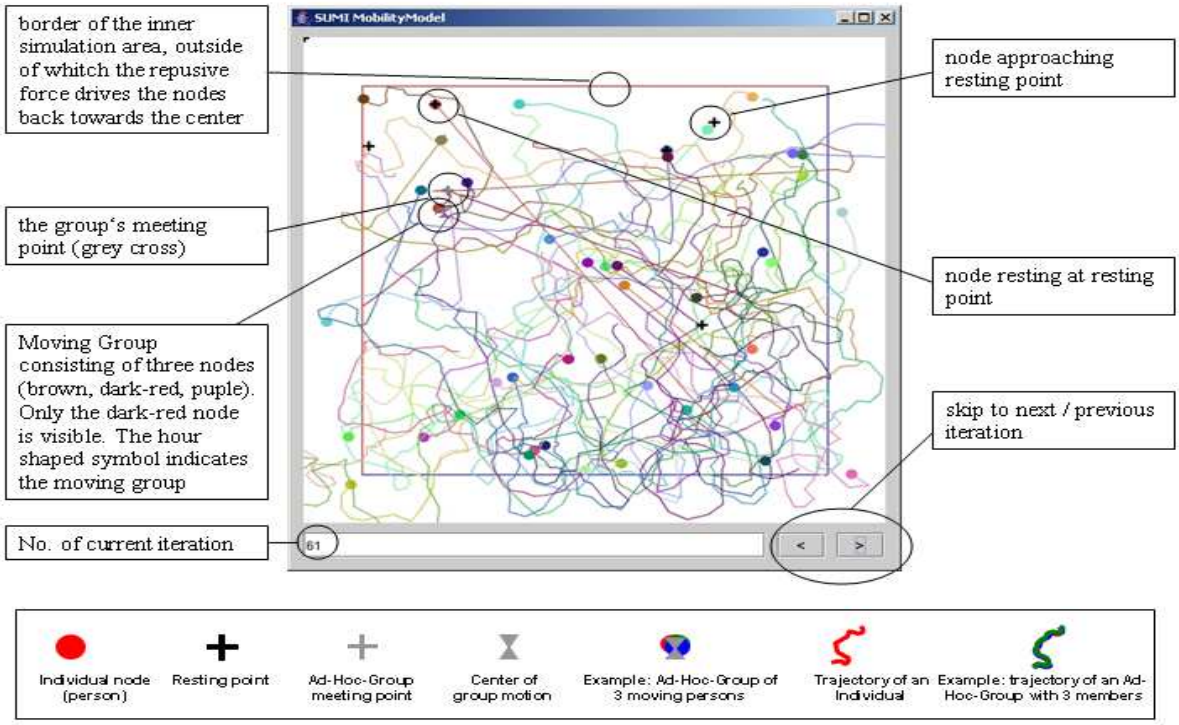


Figure 5.2: Simple Visualization of a SUMI Simulation Step. (50 nodes, $\alpha = 0.75$, area-dimensions: 500×500, number of days simulated: 3, iterations: 1024, shown iteration: no. 61, number of active groups in this iteration: 1

5.1.3.2 Basic Algorithms

We tested the three crisp clustering algorithms that were introduced in the previous sections in a testing run which involved 1024 iterations. In each iteration, the SUMI locations and

velocities were saved in a file and then groups were computed with the help of the three clustering algorithms and socially motivated and other cluster evaluation methods and the result was finally quantitatively evaluated.
As was discussed in chapter 3 and especially in section 3.10, our feature space will be what we called a **Location Phase Space** $\mathcal{L} \subseteq \mathbb{R}^4$. That means we have four dimensional feature vectors $x_k = ((x_k)_1, (x_k)_2, (x_k)_3, (x_k)_4)$ where the first two coordinates correspond to the location and the second coordinates correspond to the velocity. It is not of substantial influence whether we take the polar coordinate version of the velocity or the usual Euclidean components. In our case we took the usual Euclidean components of the velocity.
The distance metric used to compare two such patterns is the usual Euclidean distance $||\ ||$. Experiments with fuzzy clustering algorithms showed that modified distance measures like 3.39 did not give encouraging results. In combination with our socially motivated cluster validation, it was not necessary to introduce modified distance measures into our crisp clustering experiments.

5.1.3.3 Socially Motivated Cluster Validation

Usual general cluster validation measures use general cluster quality criteria (see section 5.1.2.4). Besides these general criteria, each application can specify **application specific quality measures** for clusters. In our case, this means that we will have to incorporate criteria of chapter 2 in order to determine which clusters found in location and velocity data are "good" candidates for groups.
For example a very important aspect that needs to be considered when evaluating clustering algorithms and their output is the **form of the clusters** that tends to be produced by the algorithms chosen. In case of clustering persons according to their locations and their speed it can be assumed that spherical cluster shapes represent a humans perception of a group of persons best (see chapter 2). Thus algorithms like those introduced in the previous sections are suitable for that task, because they are targeted towards finding spherical clusters [161]. There are other algorithms that are more specialized towards finding clusters with a more linear shape [161]. These algorithms should be used in the task of spatially clustering persons only in environments / situations where groups of people are likely to move in chains (as it is the case when a platoon of soldiers is moving in combat).
From the conclusions of chapter 2, some more heuristics can be derived. First, as was already explained, a group (an Ad-Hoc-Group) in a generalized spatio-temporal situation needs to have face to face contact to remain in a state of steady communication. This suggests to not only give **compact** clusters a higher score (as the conventional cluster validity measures do) but to totally discard clusters whose diameter is clearly without the limits of human perception. [2] As an order of magnitude, we presented in chapter 2 a cut-off diameter of $\tau_l = 30$ meters. In our experiments we used the more liberal bound of the simulation's equivalent of 100 meters.
The second heuristic that is socially motivated is to introduce a second cut-off maximum cluster "diameter" for the **velocities**. This motivation was extensively discussed in section 3.10. As an order of magnitude for this parameter we used a value of $\tau_v = 2.0$ (in simulation units) which was determined through quantitative evaluation (see section 5.1.3.5).
If we denote the 4 dimensional centroid vector of a cluster $\mathcal{C}_i$ with m member patterns $x_{k_1}, x_{k_2} \ldots x_{k_m}$

[2] Remember that, vice versa, it is not automatically guaranteed that every compact cluster is automatically a real Ad-Hoc-Group. With the presented techniques we can only determine **candidates** for Ad-Groups.

as

$$\pi_i = \frac{1}{m} \sum_{j=1}^{m} x_{k_j} \tag{5.11}$$

we can specify the cluster accepting criteria as

$$\begin{aligned} \forall i\ k \quad x_k \in \mathcal{C}_i \quad \rightarrow \quad & ||\big((x_k)_1, (x_k)_2\big) - \big((\pi_i)_1, (\pi_i)_2\big)|| < \tau_l \quad \wedge \\ & ||\big((x_k)_3, (x_k)_4\big) - \big((\pi_i)_3, (\pi_i)_4\big)|| < \tau_v \end{aligned} \tag{5.12}$$

In combination with the clustering algorithms shown it is the best policy to either accept a cluster as a whole or to drop the cluster as a whole. Selectively storing only the valid parts of the cluster will not improve the performance of the method, because if the group in question is not valid as a whole but has valid sub-groups, our method has a high probability of finding these in other steps of the procedure, as we will see now.

Using the social cluster validation measure we proceed as follows: For n patterns we compute (for every iteration of the simulation) clusterings with $c = 2, c = 3, \ldots, c = n - 1$ clusters. For every clustering the measure (5.12) decides which of the clusters are valid. The valid clusters from every clustering are stored and consolidated. The process of **consolidation** involves the final heuristic: If a group $\mathcal{C}_{i_1}$ is a subset of another group $\mathcal{C}_{i_1} \subset \mathcal{C}_{i_2}$ we only consider the larger group $\mathcal{C}_{i_2}$. This assumption is derived from common sense: The property of being a socially relevant Ad-Hoc-Group with respect to one quality (in our case location and velocity) is a **maximal property**: Although the group in question may have substructures with respect to other qualities (e.g. interests), humans will perceive the largest valid group with respect to one quality and not subgroups of this group. This heuristic only applies in connection with the strict criterion (5.12). If this criterion is not applied in connection with this heuristic we would introduce an artificial bias toward larger groups which is not desired.

We will call the whole Social Cluster Validation and Selection procedure **SCVS** in our experiments.

5.1.3.4 Quantitative Evaluation Measure

In order to be able to compare the results for different settings of parameters, algorithms etc., a **Precision / Recall based measure** was used. In information retrieval, the traditional measures Precision, Recall and F-Measure are based on comparing the delivered result set of an information retrieval task with the actually existing relevant set.
From the most simple point of view, starting from a query Q and a database $\mathcal{D}$ an **information retrieval system** delivers an answer set $\mathcal{A} \subseteq \mathcal{D}$. The database is an unordered but enumerable set of atomic items $\mathcal{D} = \{d_1, d_2, \ldots, d_{|\mathcal{D}|}\}$. The same applies to the answer set: $\mathcal{A} = \{a_1, a_2, \ldots, a_{|\mathcal{A}|}\}$. Furthermore we have a binary relevance function $R_Q : \mathcal{D} \rightarrow \{0, 1\}$ that decides for every element in the database (including the answer set) whether it is relevant with respect to the query or not.
We can now define four numbers that quantitatively characterize the quality of the answer set:

- **True Positives TP**: The number of relevant elements of $\mathcal{A}$
  That is $|\{a_k \in \mathcal{A} | R_Q(a_k) = 1\}|$.

- **False Positives FP**: The number of irrelevant elements of $\mathcal{A}$
  That is $|\{a_k \in \mathcal{A} | R_Q(a_k) = 0\}|$.
- **True Negatives TN**: The number of irrelevant elements of $\mathcal{D} \setminus \mathcal{A}$
  That is $|\{d_k \in \mathcal{D} \setminus \mathcal{A} | R_Q(d_k) = 0\}|$.
- **False Negatives TN**: The number of relevant elements of $\mathcal{D} \setminus \mathcal{A}$
  That is $|\{d_k \in \mathcal{D} \setminus \mathcal{A} | R_Q(d_k) = 1\}|$.

The traditional measures are then defined as:

- **Precision P**:

$$P = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{5.13}$$

- **Recall R**:

$$R = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{5.14}$$

- **F-Measure**:

$$F_\alpha = (\alpha P^{-1} + (1 - \alpha) R^{-1})^{-1} \tag{5.15}$$

Transferring this to our problem of identifying Ad-Hoc-Groups of a SUMI simulation with respect to locations and velocities we have the following situation:
Let the superset $\mathcal{G}^{(r)}$ of "real" groups (groups that are simulated as such by SUMI) in an iteration step be denoted as:

$$\mathcal{G}^{(r)} = \{\mathcal{G}_1^{(r)}, \mathcal{G}_2^{(r)}, \ldots, \mathcal{G}_{|\mathcal{G}^{(r)}|}^{(r)}\}$$

This includes random groups as well as periodic groups, moving groups as well as resting groups.
Let the superset $\mathcal{G}^{(f)}$ of found groups (clusters) in the same iteration step be denoted as:

$$\mathcal{G}^{(f)} = \{\mathcal{C}_1^{(f)}, \mathcal{C}_2^{(f)}, \ldots, \mathcal{C}_{|\mathcal{G}^{(f)}|}^{(f)}\}$$

If we assume that we can only compare real groups $\mathcal{G}_{i_1}^{(r)}$ with clusters $\mathcal{C}_{i_2}^{(f)}$ in terms of identity or non-identity:

$$s_{i_1 i_2} = \mathrm{sim}(\mathcal{G}_{i_1}^{(r)}, \mathcal{C}_{i_2}^{(f)}) \in \{0, 1\} \tag{5.16}$$

(which is the equivalent of the relevance function R_Q) and if we do not have duplicates in either $\mathcal{G}^{(r)}$ or $\mathcal{G}^{(f)}$ we have the condition that:

$$\sum_{i_1=1}^{|\mathcal{G}^{(r)}|} s_{i_1 i_2} \overset{!}{\leq} 1 \quad \text{and} \quad \sum_{i_2=1}^{|\mathcal{G}^{(f)}|} s_{i_1 i_2} \overset{!}{\leq} 1 \tag{5.17}$$

Thus we get

$$\text{TP} = \sum_{i_1=1}^{|\mathcal{G}^{(r)}|} \sum_{i_2=1}^{|\mathcal{G}^{(f)}|} s_{i_1 i_2} \qquad (\#\text{ of 1s in the matrix}) \tag{5.18}$$

$$\text{FP} = \sum_{i_2=1}^{|\mathcal{G}^{(f)}|} (1 - \sum_{i_1=1}^{|\mathcal{G}^{(r)}|} s_{i_1 i_2}) \qquad (\#\text{ of columns with only 0s}) \tag{5.19}$$

$$\text{FN} = \sum_{i_1=1}^{|\mathcal{G}^{(r)}|} (1 - \sum_{i_2=1}^{|\mathcal{G}^{(f)}|} s_{i_1 i_2}) \qquad (\#\text{ of rows with only 0s}) \tag{5.20}$$

which simply leads to

$$P = \frac{\text{TP}}{|\mathcal{G}^{(f)}|} \tag{5.21}$$

$$R = \frac{\text{TP}}{|\mathcal{G}^{(r)}|}. \tag{5.22}$$

We can apply the same considerations to the comparison of a **single group** $\mathcal{G}_{i_1}^{(r)}$ with a single **found cluster** $\mathcal{C}_{i_2}^{(f)}$. Thus we are able to do more than just compare these two sets in terms of identity or non-identity as in equation (5.16). We will replace the simple binary similarity measure of equation (5.16) with the $F_{0.5}$ **measure**:
Denoting the set of nodes (patterns, simulated people) like in the sections before by $\mathcal{X} = \{x_1, x_2 \ldots, x_{|\mathcal{X}|}\}$, the real group by

$$\mathcal{G}_{i_1}^{(r)} = \{x_{k_1}^{(r)}, x_{k_2}^{(r)}, \ldots, x_{k_{|\mathcal{G}_{i_1}^{(r)}|}}^{(r)}\}$$

and the found cluster by

$$\mathcal{C}_{i_2}^{(f)} = \{x_{k_1}^{(f)}, x_{k_2}^{(f)}, \ldots, x_{k_{|\mathcal{C}_{i_2}^{(f)}|}}^{(f)}\}$$

we can use as a continuous similarity measure between $\mathcal{G}_{i_1}^{(r)}$ and $\mathcal{C}_{i_2}^{(f)}$ the $F_{0.5}$ measure that results from comparing its members via the simple identity measure

$$\tilde{s}_{k_l^{(r)}, k_m^{(f)}} = \text{sim}(x_{k_l}^{(r)}, x_{k_m}^{(f)}) = \delta_{k_l^{(r)}, k_m^{(f)}} \in \{0, 1\} \tag{5.23}$$

Inserting the analog of the expressions (5.21) and (5.22) into the definition of the F-measure(5.15) (with $\alpha = \frac{1}{2}$) gives

$$[0,1] \ni s_{i_1 i_2} = \text{sim}(\mathcal{G}_{i_1}^{(r)}, \mathcal{C}_{i_2}^{(f)}) = \left(\frac{1}{2} \frac{\sum_{m=1}^{|\mathcal{G}_{i_1}^{(r)}|} \sum_{l=1}^{|\mathcal{C}_{i_2}^{(f)}|} \tilde{s}_{k_l^{(r)}, k_m^{(f)}}}{|\mathcal{C}_{i_2}^{(f)}|} + \frac{1}{2} \frac{\sum_{m=1}^{|\mathcal{G}_{i_1}^{(r)}|} \sum_{l=1}^{|\mathcal{C}_{i_2}^{(f)}|} \tilde{s}_{k_l^{(r)}, k_m^{(f)}}}{|\mathcal{G}_{i_1}^{(r)}|} \right)^{-1} \tag{5.24}$$

This F-Measure based expression for the similarity between two sets of patterns is very similar to the Hamming-Distance based similarity measure which we will use later in this chapter. This

F-Measure based similarity measure was favored instead of the symmetric Hamming Distance based measure because of the "classical" IR-like asymmetric perspective of the task of comparing a "given" group structure with a "found" cluster structure.
In order to compare the found and real groups we also have to take in account at each iteration, what part of the scheduled group has already reached the group's meeting point. These quantities can readily be computed from simulation data.
Since the expressions (5.16) and (5.17) do not hold any longer in the case of a continuous similarity measure $\mathrm{sim}(\mathcal{G}_{i_1}^{(r)}, \mathcal{C}_{i_2}^{(f)}) \in [0,1]$ defined in equation (5.24), we need to modify the similarity matrix in order to keep our expressions (5.21) and (5.22) for the overall precision and overall recall of one iteration.
In order to force the fulfillment of the conditions (5.17) we will modify the matrix $s_{i_1 i_2}$ computed with equation (5.24). The modification first sets all elements in a row to zero except the maximum element of the row and then sets all elements in a column to zero except the maximum element of the column:

$$\bar{s}_{i_1 i_2} = \begin{cases} s_{i_1 i_2} & \text{if} \quad s_{i_1 i_2} = \max_i s_{i_1 i} \\ 0 & \text{else} \quad . \end{cases} \tag{5.25}$$

$$\bar{\bar{s}}_{i_1 i_2} = \begin{cases} \bar{s}_{i_1 i_2} & \text{if} \quad \bar{s}_{i_1 i_2} = \max_i \bar{s}_{i i_2} \\ 0 & \text{else} \quad . \end{cases} \tag{5.26}$$

The philosophy behind this process is that although a real group may have several similar found clusters and vice versa we will count only the most similar counterpart. A very similar technique has been applied in section 4.7.2 when discussing the comparison of unordered sets of interest phrases (see equation (4.5)).
Replacing s with $\bar{\bar{s}}$ in the equation (5.18) for the true positives TP we can use equations (5.21) and (5.22) to compute the precision and recall for one iteration. If no groups exist but clusters are found by the algorithm or groups exist but no clusters are found we set $R = P = 0$. If no groups exist and no clusters are found we set $R = P = 1$.
We average the found values of R, P and $F_{0.5}$ over all iterations and receive total quality measures R_{tot}, P_{tot} and $F_{0.5\mathrm{tot}}$ for the complete simulation. We will now discuss the results that were achieved with varying the algorithm's parameters.

5.1.3.5 Results

For the following experiments a single SUMI simulation of 1024 iterations was used (for the other data of the simulation see figure 5.2). We will first take a look at how the choice of the clustering algorithm affects the performance of the overall procedure.

Algorithms We will use our social cluster validation technique and proceed as described in the previous sections. First we computed a 1024 iteration SUMI mobility simulation with standard parameters. These simulation data have been used throughout all experiments.
Varying the cluster algorithm yielded the results depicted in table 5.1 The first thing that is apparent from table 5.1 is that the performance of our procedure of finding groups on the basis of location and speed data is excellent. We are able to find the majority of all simulated groups with high precision. This can mostly be attributed to our **SCVS** procedure. We will discuss this issue later in more depth.

	SAHN	K-Means	MST
Precision P_{tot}	0.8283	0.7843	0.8283
Recall R_{tot}	0.7111	0.6663	0.7111
F-Measure $F_{0.5\text{tot}}$	0.7523	0.7073	0.7523
Computing Time [sec]	367.5	150.5	11910.7

Table 5.1: Varying the Clustering Algorithm. SAHN was single link. A Euclidean distance measure and the Social Cluster Validation and Selection procedure **SCVS** were used for all three experiments. The same SUMI simulation data was used for all three experiments. SCVS parameters were $\tau_l = 1.0$ and $\tau_v = 2.0$

What we can see further is that varying the core clustering algorithm does not influence the precision and recall values substantially. The K-Means variant slightly shows the well known tendency to converge to local optima and performs marginally worse. SAHN and MST produce identical precision and recall values which is an indication that the clustering results delivered by the two algorithms do not differ substantially enough to give different clusters after the **SCVS** procedure. The MST is inferior with respect to runtime. That's why we kept the K-Means algorithm for most of the rest of our experiments.

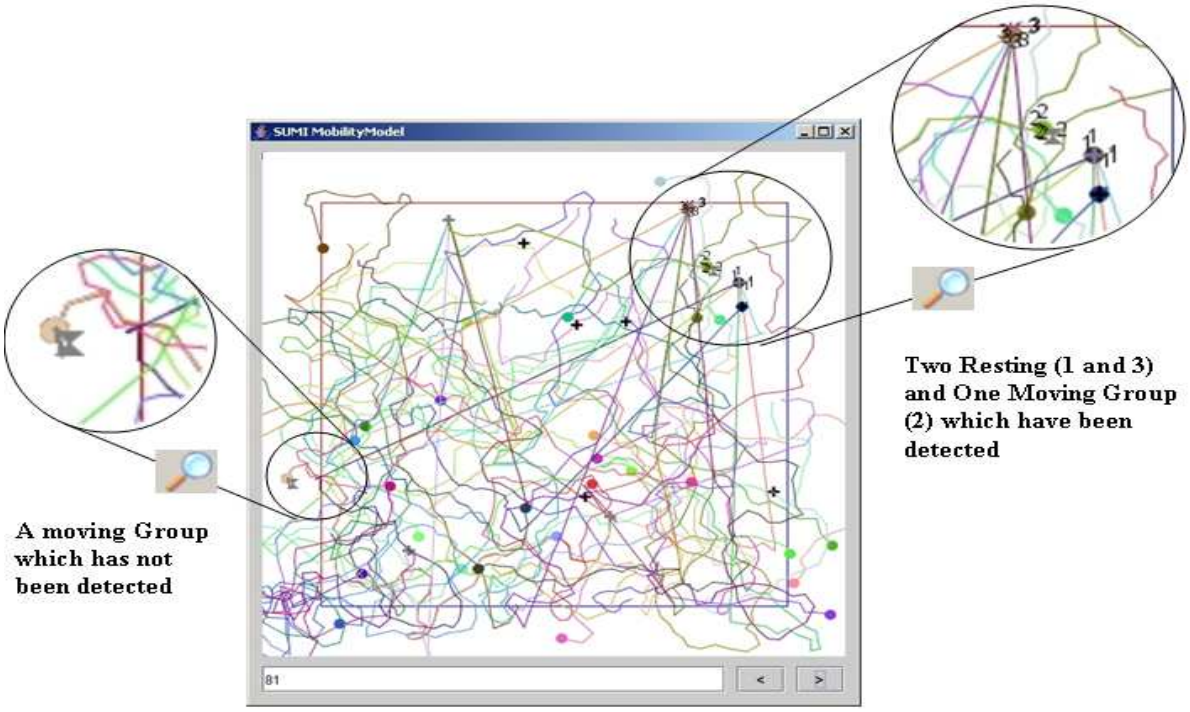


Figure 5.3: Simple Visualization of a SUMI Simulation Step with SCVS Ad-Hoc-Group detection. (50 nodes, $\alpha = 0.75$, area-dimensions: 500×500, number of days simulated: 3, iterations: 1024, shown iteration: no. 81, number of active groups in this iteration: 4. Number of found groups: 3

Distance Measures and Cluster Validation & Selection In section 3.10 it was discussed, how altering the similarity measure or the corresponding **distance measure** between two spatio-temporal situation vectors could be a means to incorporate the fact that larger distances lead

to a substantial decrease of social relatedness. We have already discussed in chapter 2, that several factors in human social psychology lead to this decrease. In section 5.1.3.3 above we have introduced a cluster validation and selection procedure (**SCVS**) that acts as a way to socially filter the results delivered by the plain clustering algorithms.
To enlighten the correlations between the distance measure used on the one hand and the **cluster validation** (and selection) strategy on the other hand, these influence factors have been systematically evaluated.
First, we will vary the distance measure. Section 3.10 introduced a similarity measure based on an exponential decrease with increasing distance. We will call the corresponding distance measure **Exponential Distance**.
Based on the **Euclidean Distance**

$$d(x_{k_1}, x_{k_2}) = ||x_{k_1} - x_{k_2}||$$

the expression for the **Exponential Distance** is

$$d(x_{k_1}, x_{k_2}) = \begin{cases} ||x_{k_1} - x_{k_2}|| & \text{if} \quad d(x_{k_1}, x_{k_2}) \leq a \\ a + \Big(||x_{k_1} - x_{k_2}|| - a\Big) \mathrm{e}^{||x_{k_1} - x_{k_2}|| - a} & \text{if} \quad d(x_{k_1}, x_{k_2}) > a \end{cases} . \tag{5.27}$$

We have also tested a less strict version of the distance boosting factor. Instead of the exponential function we incorporated a polynomial growth (**Polynomial Euclidean Distance**):

$$d(x_{k_1}, x_{k_2}) = \begin{cases} ||x_{k_1} - x_{k_2}|| & \text{if} \quad d(x_{k_1}, x_{k_2}) \leq a \\ a + \Big(||x_{k_1} - x_{k_2}|| - a\Big)^3 & \text{if} \quad d(x_{k_1}, x_{k_2}) > a \end{cases} . \tag{5.28}$$

We varied the "cut-off" distance a in several steps. Remember from chapter 3 that the inter-group-movement parameter γ in SUMI was set to the simulation's equivalent of 30 metres. (This parameter controls the "diameter" of moving or resting groups (periodic as well as random) (see section 3.7)). In simulation units (assuming a 500×500 simulation area corresponding to 25 km $\times$ 25 km, the value of 30 meters corresponds to a value of 0.6.

		Euclidean	Exponential Euclidean			Polynomial Euclidean		
SCVS	a	-	0.5	1	3	0.5	1	3
	Precision P_{tot}	0.7843	0.7843	0.7843	0.7843	0.7843	0.7843	0.7843
	Recall R_{tot}	0.6663	0.6663	0.6663	0.6663	0.6663	0.6663	0.6663
	F-Measure $F_{0.5\text{tot}}$	0.7073	0.7073	0.7073	0.7073	0.7073	0.7073	0.7073
	Comp. Time [sec]	150.5	210.6	220.2	215.3	179.2	181.3	174.5
Dunn	a	-	0.5	1	3	0.5	1	3
	Precision P_{tot}	0.2451	0.2451	0.2451	0.2451	0.2451	0.2451	0.2451
	Recall R_{tot}	0.2451	0.2451	0.2451	0.2451	0.2451	0.2451	0.2451
	F-Measure $F_{0.5\text{tot}}$	0.2451	0.2451	0.2451	0.2451	0.2451	0.2451	0.2451
	Comp. Time [sec]	162.7	222.22	210.2	215.6	160.6	164.6	161.5

Table 5.2: Varying the Distance Measure. K-Means was used for all experiments and the same SUMI simulation data was used for all experiments. We used the Social Cluster Validation and Selection procedure **SCVS** and the Dunn Index in order to compare the results with respect to cluster validation and selection. SCVS parameters were $\tau_l = 1.0$ and $\tau_v = 2.0$

The first thing that is apparent from table 5.2 is that the introduction of an **altered distance measure** does **not** appear to **change the clustering results** so that the precision and recall

values are also not changed. This can be attributed to the nature of crisp clustering algorithms which are obviously not very sensitive to these changes in distance measures. In the experiments with fuzzy clustering of location and speed data the clustering results were much more sensitive to changing the distance measure. As a consequence, the normal Euclidean measure should be preferred and more emphasis should be put on the cluster evaluation and selection strategy.
The most important thing that can be learned from the experiments in table 5.2 is that our Social Cluster Validation and Selection procedure **SCVS** performs **significantly better** than conventional cluster validity criteria while being not significantly more complex with respect to computation time. This is a very remarkable result that backs up our considerations of the previous sections and chapters.
While obtaining good results with the described procedures, the **critique** that **retrofitting** an analysis procedure to a specific structure of the data simulation may be scientifically **arguable** should be taken seriously. What can be replied to this critique is that the simulation process is thoroughly tailored to deliver data that are as realistic as possible, using many heuristics that are motivated from social psychology. Furthermore, the analysis procedure is independent of the simulation and uses no direct knowledge of the simulation process. Thus, we have the confidence that the quality of the results is not artificial. In order to systematically investigate hidden relations between the simulation and the analysis procedure, we would have to conduct tests with real data, where data about the real group structures would have to be collected independently which is a very complicated task.

Selection Parameters We will now investigate how the **selection parameters** τ_l and τ_v of the **SCVS** procedure influence the results of the group detection process. The parameters (see equation (5.12) for their exact specification) determine the limits which decide whether to accept a cluster as an Ad-Hoc-Group or not. We have introduced two such parameters to be able to influence the behavior of the selection separately for locations and velocities.
Varying both parameters systematically, yielded the results given in table 5.3. Figure 5.4 graphically shows the results (the F-Measure $F_{0.5}$) of **varying the location selection parameter** τ_l while keeping the velocity selection parameter τ_v constant. The behavior of the curves is in coincidence with the expectations. If we choose a too restricted velocity selection (e.g. $\tau_v = 0.2$) the precision and recall values are very low no matter what the location selection parameter is chosen to be. The other curves have the expected behavior: if we continuously widen the area corresponding to allowed cluster diameters, the precision and recall values start to increase, reach a maximum and fall back to very low values if τ_l is getting too big. This basic behavior is in accordance with our heuristics: if we are too strict with judging a cluster's appropriateness as a group we will not detect all groups (low recall) or only parts of the groups (low precision (remember that we have implemented a gradual version of similarity (see section 5.1.3.4))). If we are getting to lax we also accept clusters as groups which are not in accordance with our social heuristics. Since the simulation is also computed with respect to these heuristics, we will get a rather inferior precision.
Varying the velocity selection parameter gives the results depicted in figure 5.5. Although we find the same τ_v behavior for reasonable values of τ_l (e.g. $\tau_l = 5.0$) as in the case of varying τ_l, the curves in the lower half of the figure are not in complete coincidence with our expectations. For a value of $\tau_l = 1.0$ we find a steady increase in precision and recall which starts to slightly decrease again only at values of $\tau_v = 50.0$ and higher (Not shown in the figures and in table 5.3). We can only explain this behavior by an undetected error in the simulation's computation of the speed values. Basically, we would expect a decrease in precision if we loosen the velocity

τ_l	0.2	0.2	0.2	0.2	0.2	0.2	0.2
τ_v	0.2	0.5	1.0	3.0	5.0	10.0	30.0
Precision P_{tot}	0.28	0.29	0.32	0.32	0.32	0.31	0.31
Recall R_{tot}	0.28	0.30	0.34	0.36	0.36	0.36	0.37
F-Measure $F_{0.5\,\text{tot}}$	0.28	0.29	0.32	0.33	0.33	0.33	0.33
τ_l	0.5	0.5	0.5	0.5	0.5	0.5	0.5
τ_v	0.2	0.5	1.0	3.0	5.0	10.0	30.0
Precision P_{tot}	0.26	0.30	0.62	0.72	0.74	0.76	0.79
Recall R_{tot}	0.25	0.29	0.56	0.64	0.66	0.73	0.81
F-Measure $F_{0.5\,\text{tot}}$	0.26	0.29	0.58	0.66	0.70	0.74	0.80
τ_l	1.0	1.0	1.0	1.0	1.0	1.0	1.0
τ_v	0.2	0.5	1.0	3.0	5.0	10.0	30.0
Precision P_{tot}	0.26	0.30	0.68	0.79	0.82	0.84	0.88
Recall R_{tot}	0.25	0.29	0.60	0.68	0.73	0.79	0.87
F-Measure $F_{0.5\,\text{tot}}$	0.25	0.29	0.62	0.72	0.76	0.81	0.87
τ_l	3.0	3.0	3.0	3.0	3.0	3.0	3.0
τ_v	0.2	0.5	1.0	3.0	5.0	10.0	30.0
Precision P_{tot}	0.26	0.30	0.67	0.77	0.78	0.75	0.69
Recall R_{tot}	0.25	0.29	0.58	0.67	0.70	0.72	0.75
F-Measure $F_{0.5\,\text{tot}}$	0.25	0.29	0.61	0.71	0.72	0.72	0.70
τ_l	5.0	5.0	5.0	5.0	5.0	5.0	5.0
τ_v	0.2	0.5	1.0	3.0	5.0	10.0	30.0
Precision P_{tot}	0.25	0.30	0.65	0.74	0.73	0.64	0.52
Recall R_{tot}	0.24	0.28	0.57	0.65	0.66	0.64	0.65
F-Measure $F_{0.5\,\text{tot}}$	0.25	0.28	0.59	0.68	0.68	0.62	0.55
τ_l	10.0	10.0	10.0	10.0	10.0	10.0	10.0
τ_v	0.2	0.5	1.0	3.0	5.0	10.0	30.0
Precision P_{tot}	0.22	0.24	0.54	0.59	0.53	0.40	0.32
Recall R_{tot}	0.22	0.23	0.48	0.53	0.52	0.50	0.58
F-Measure $F_{0.5\,\text{tot}}$	0.22	0.23	0.49	0.54	0.50	0.42	0.38
τ_l	30.0	30.0	30.0	30.0	30.0	30.0	30.0
τ_v	0.2	0.5	1.0	3.0	5.0	10.0	30.0
Precision P_{tot}	0.16	0.13	0.32	0.31	0.27	0.16	0.10
Recall R_{tot}	0.15	0.12	0.33	0.36	0.38	0.37	0.38
F-Measure $F_{0.5\,\text{tot}}$	0.15	0.12	0.31	0.32	0.29	0.20	0.14

Table 5.3: Varying the SCVS parameters τ_l and τ_v. K-Means and the same SUMI simulation data were used for all experiments. We used Euclidean distance and (of course) the Social Cluster Validation and Selection procedure **SCVS**.

selection policy too much. Since we added a random component to the velocity vectors of group members, the speed vectors might show a larger diversity within a group than it is the case with the location vectors. Imagine a group practicing basketball together where a certain diversity in speed vectors seems natural. What we can thus learn from this experiment is that the location selection parameter is likely to much more sensitively influence the performance of the algorithm

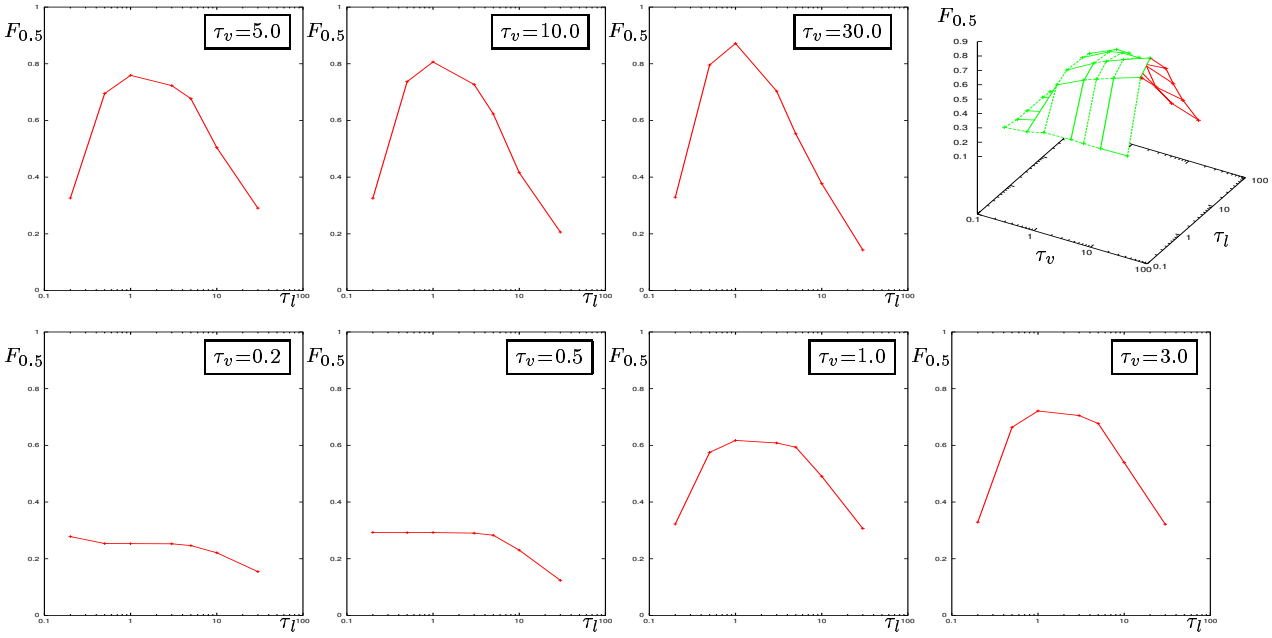


Figure 5.4: Varying the SCVS parameters τ_v and τ_l. K-Means and the same SUMI simulation data were used for all experiments. We used Euclidean distance and (of course) the Social Cluster Validation and Selection procedure **SCVS**. The parameter τ_l is shown in logarithmic scale in the 2-d-plots and both parameters are shown in log-scale in the 3d-plot.

than the speed selection parameter.

5.1.4 Detecting and Modeling Abstract Groups

What we have discussed in the previous sections is the detection and modeling of Ad-Hoc-Groups based on locations and velocities. In chapter 2 it was discussed that Ad-Hoc-Groups can be perceived as instantiations of **abstract groups**. This point of view is very important in identifying Ad-Hoc-Groups as clusters which have a high degree of **social relevance**. As an example it was discussed that people standing in line at a fast food restaurant could be wrongly identified by our procedure so far as an Ad-Hoc-Group with respect to location and velocity. In most cases, the social relevance of this group is low if not zero. These groups have been called **pseudo groups** in chapter 2. In order to target the discussed algorithms towards finding only socially relevant Ad-Hoc-Groups and in order to improve the scope of the algorithms, we need to analyze the underlying **abstract group structure** as well. In order to do so, we need to develop and algorithmically use heuristics that decide which Ad-Hoc-Groups are likely to be instantiations of an abstract group and which are not. Among these heuristics, two are of special importance and will be investigated in more detail here. The first criterion is that the Ad-hoc-Groups which are associated with an abstract group should be **similar with respect to members**. The second heuristic aims at the regularity of the meeting pattern: If a set of similar Ad-Hoc-Groups occurs in **regular time intervals** the probability that this set is considered an abstract group is high. These aspects will now be discussed in more detail while presenting the detection and modeling algorithm step by step.

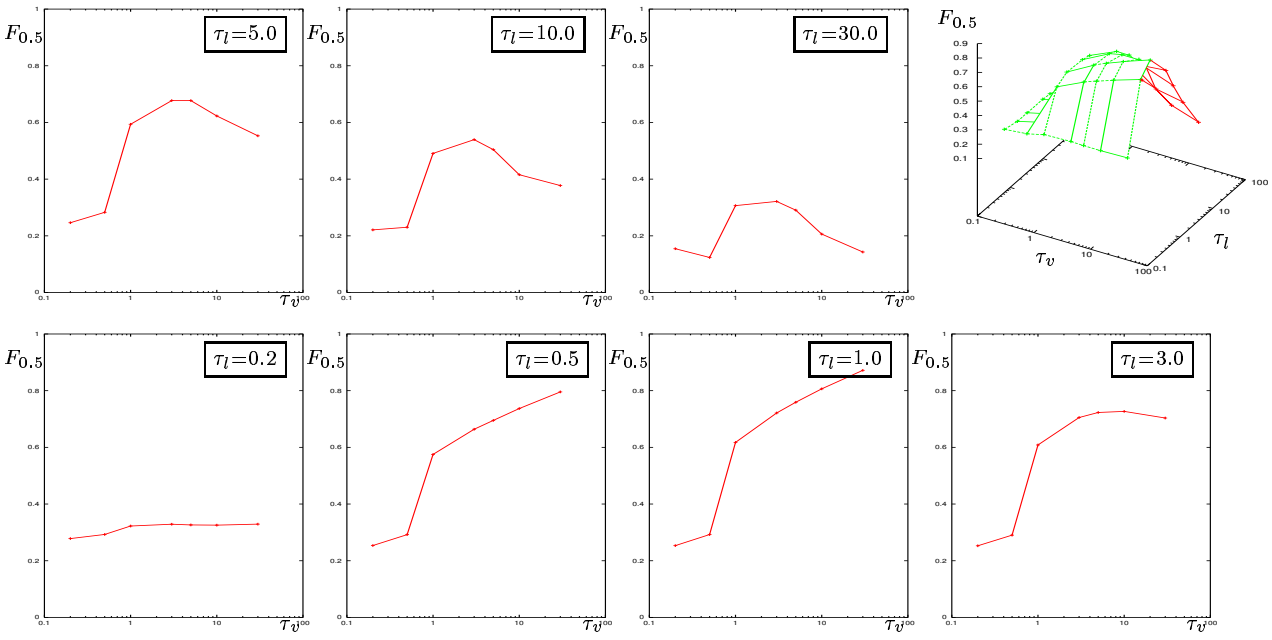


Figure 5.5: Varying the SCVS parameters τ_v and τ_l. K-Means and the same SUMI simulation data were used for all experiments. We used Euclidean distance and (of course) the Social Cluster Validation and Selection procedure **SCVS**. The parameter τ_v is shown in logarithmic scale in the 2-d-plots and both parameters are shown in log-scale in the 3d-plot.

5.1.4.1 Step 1: Similarity with respect to members

When trying to compare Ad-Hoc-Groups with one another with the aim of determining their association with an abstract group, the first thing to assess is the problem of determining a **unique identifier** for an Ad-Hoc-Group. In case of crisp clusters (groups) this problem (as well as the problem of stability) is not as difficult as in the case of fuzzy clusters (see section 5.2). As a unique identifier of an Ad-Hoc-Group, we will use the set of its members. If any algorithm or software system needs to map this identifier-set to a number, this can easily by done with the help of a Hash-function, e.g. through a prime number Goedelization. Usually we will denote the index (numerical identifier of a group) with i.
In a platform oriented community setting, the members usually have distinct identifiers (e.g. member-id-numbers). If we regard the case of a distributed community support system, we might use other means of unique identifiers which also can be mapped to numbers via hashing. (we will denote the index (numerical identifier of a person (pattern)) with k).Thus we will assume that member-ids are natural numbers (index k) and that Ad-Hoc-Group identifiers are sets of natural numbers. The sets as are counted by a single index i.
In order to decide the degree of **similarity between two crisp Ad-Hoc-Groups**, we can rely on well known distance measures. In contrast to the IR-motivated, a priori asymmetric measures that have been used for computing the quality measures for our group detection and modeling procedure (see section 5.1.3.4) we are now interested in a symmetric similarity measure.

Denoting the two groups to be compared as

$$\begin{aligned}\mathcal{G}_{i_1} &= \{x_{k_1}^{(i_1)}, x_{k_2}^{(i_1)}, \ldots, x_{k_{|\mathcal{G}_{i_1}|}}^{(i_1)}\} \\ \mathcal{G}_{i_2} &= \{x_{k_1}^{(i_2)}, x_{k_2}^{(i_2)}, \ldots, x_{k_{|\mathcal{G}_{i_2}|}}^{(i_2)}\}\end{aligned}$$

we transfer our notion of true and false positives and true and false negatives presented in section 5.1.3.4 into the language of symmetric distance measures. The true positives TP are the number of elements that both sets have in common, the false positives FP and false negatives FN are the number of elements that the first set contains but the second set does not contain (or vice versa) and the (irrelevant) quantity true negatives TN is the number of elements that neither of the two sets contains. Since we are interested in symmetric similarity measures, it is not important in which of the two ways FP and FN are defined. We will also seek to avoid the incorporation of TN because if the number of possible members is very large compared to the size of the groups, TN will also be large and of small informational value.

$$a = \text{TP} = |\{k_m | x_{k_m} \in \mathcal{G}_{i_1} \wedge x_{k_m} \in \mathcal{G}_{i_2}\}| \tag{5.29}$$
$$b = \text{FP} = |\{k_m | x_{k_m} \in \mathcal{G}_{i_1} \wedge x_{k_m} \notin \mathcal{G}_{i_2}\}| \tag{5.30}$$
$$c = \text{FN} = |\{k_m | x_{k_m} \notin \mathcal{G}_{i_1} \wedge x_{k_m} \in \mathcal{G}_{i_2}\}| \tag{5.31}$$
$$d = \text{TN} = |\{k_m | x_{k_m} \notin \mathcal{G}_{i_1} \wedge x_{k_m} \notin \mathcal{G}_{i_2}\}| \tag{5.32}$$

Based on a, b and c we can define several well known distance and similarity measures:

- **Hamming distance**

$$d(\mathcal{G}_{i_1}, \mathcal{G}_{i_2}) = b + c \tag{5.33}$$

- **Hamming distance based Similarity**

$$\text{sim}(\mathcal{G}_{i_1}, \mathcal{G}_{i_2}) = (1 + (b + c))^{-1} \tag{5.34}$$

- **Simple Matching Coefficient**

$$\text{sim}(\mathcal{G}_{i_1}, \mathcal{G}_{i_2}) = 1 - \frac{1}{|\mathcal{X}|(b + c)} \tag{5.35}$$

- $F_{0.5}$ **Measure**

$$\text{sim}(\mathcal{G}_{i_1}, \mathcal{G}_{i_2}) = (1 + \frac{b + c}{2a})^{-1} \tag{5.36}$$

The equation for the $F_{0.5}$ **Measure** is identical to equation (5.15) using the notations (5.29) - (5.32).
The Hamming distance based similarity measure (5.34) solely emphasizes the differences between the two groups no matter what size they have. Thus two ten member groups with only two differing members each would have the same distance (similarity) than two three members groups with only one common member. Thus the measure of (5.34) was thus modified by adding a term which regards the number of common members in relation to the size of the two groups:

$$\text{sim}(\mathcal{G}_{i_1}, \mathcal{G}_{i_2}) = \frac{1}{w_1 + w_2}(w_1(1 + (b + c)^{-1})^{-1} + w_2 \frac{2a}{|\mathcal{G}_{i_1}| + |\mathcal{G}_{i_2}|}|) \tag{5.37}$$

This measure delivers almost the identical values as the $F_{0.5}$ measure (equation (5.36)) if the weights are adjusted as $w_1 = 0.3$ and $w_2 = 0.7$.
With the help of these similarity measures we are able to compare two found Ad-Hoc-Groups with respect to their members.

5.1.4.2 Step 2 Extraction of Periodicity Information

Besides the similarity with respect to members, it is an important information whether the Ad-Hoc Groups are also formed in regular time-intervals. Both aspects together are a strong indication that the Ad-Hoc-Groups are instantiations of an abstract group. We will thus investigate, how the **periodicity information** can be extracted.
If we plot the occurrences of a single Ad-Hoc Group over the discrete time steps that the group detection procedure is carried out at, we might get a "binary signal" like in the upper part of figure 5.6. The problem with those "signals" from single Ad-Hoc-Groups is that it usually shows

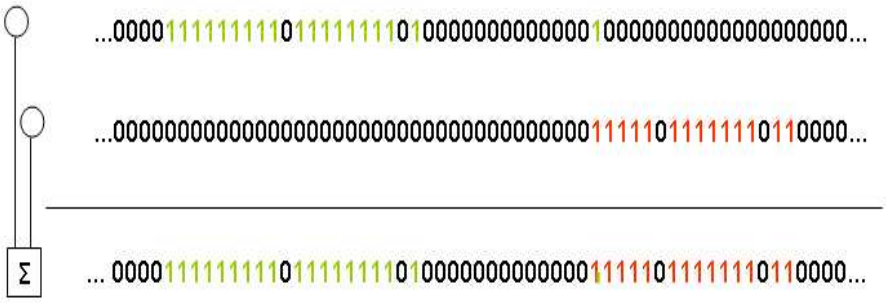


Figure 5.6: The occurrences of two Ad-Hoc-Groups over several subsequent runs of the group detection procedure. In our testing scenario these points in time correspond to a single iteration of the SUMI simulation. "1" stands for "the group exists" (has been detected) and "0" stands for "the group does not exist". The last row shows the union of the two "signals".

only **isolated occurrences** of this group because it is not very likely that the exact same group of persons will meet again. E.g. in the scenario of the practicing basketball team, there might be different persons missing at each practice which gives different groups at each practice. We have already shortly discussed these issues in section 3.8. The consequence is that, in order to be able to detect the frequency of instantiation of an abstract group we need to **unite the time-signals** of those groups which are likely to be instantiations of this single abstract group. As was discussed in the previous section, a good criterion is the **similarity with respect to the members**. So the signals of all the groups that are above a similarity threshold when compared to a group in question are united and then the period is computed.
We denote the time signal of a group $\mathcal{G}_i = \{x_{k_1}^{(i)}, x_{k_2}^{(i)}, \ldots, x_{k_{|\mathcal{G}_i|}}^{(i)}\}$ by a discrete time series f:

$$f_k^{(i)} = f^{(i)}(t_k) = \begin{cases} 1 & \text{if } \mathcal{G}_i \text{was detected at time } t_k \\ 0 & \text{if } \mathcal{G}_i \text{was not detected at time } t_k \end{cases} \tag{5.38}$$

The process of uniting a group's time signal with the signals of its most similar groups is then formally stated as

$$f^{(i)} \rightarrow f_{\Sigma}^{(i)} = f^{(i)} + \sum_{\{i_m | \mathrm{sim}(\mathcal{G}_i, \mathcal{G}_{i_m}) > p \wedge i \neq i_m\}} f^{(i_m)} = \sum_{\{i_m | \mathrm{sim}(\mathcal{G}_i, \mathcal{G}_{i_m}) > p\}} f^{(i_m)} \tag{5.39}$$

where p denotes the similarity threshold.
Having synthesized a probable time series for the abstract group, we can now analyze this time series in order to find the period (if any) with that this abstract group manifests itself in time.

5.1.4.3 Fourier Analysis

A well known tool for analyzing the frequency spectrum of a time signal that is used in all areas of science is Fourier Analysis. If we have a continuous signal $f(t)$ it's Fourier-Transform is defined by

$$(\mathscr{F} f)(\nu) = \int_{-\infty}^{\infty} f(t) \mathrm{e}^{-\mathrm{i}2\pi\nu t} \mathrm{d}t \quad . \tag{5.40}$$

If we transform a function f of the form

$$f_b(t) = \begin{cases} 1 & \text{if} \quad |t| \leq b \\ 0 & \text{if} \quad \text{else} \end{cases} \tag{5.41}$$

we easily get:

$$(\mathscr{F} f_b)(\nu) = \int_{-b}^{b} \mathrm{e}^{-\mathrm{i}2\pi\nu t} \mathrm{d}t = \frac{1}{\pi\nu} \sin 2\pi b\nu \tag{5.42}$$

We will now derive how the Fourier transform of a function looks like which is the sum of

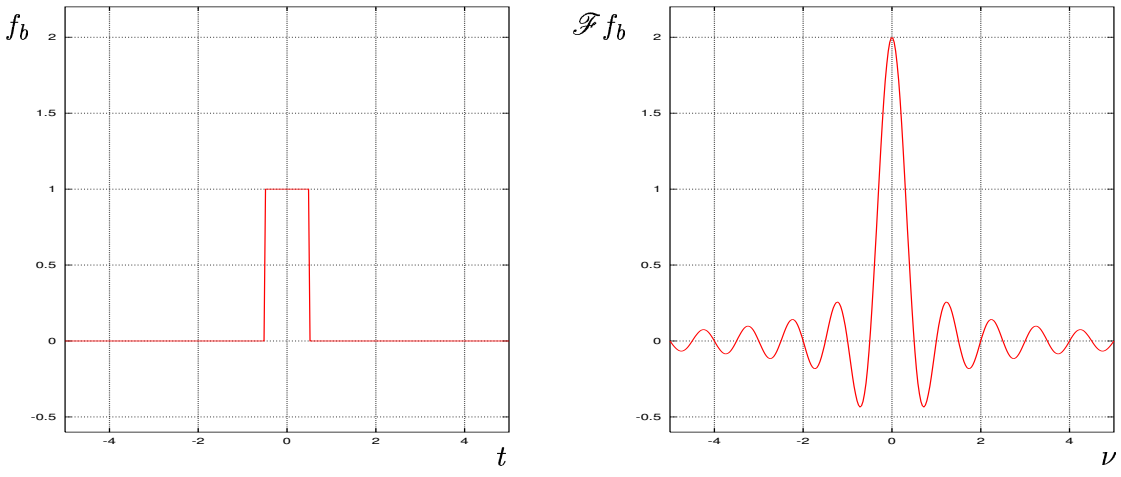


Figure 5.7: The function f_b and its Fourier transform $\mathscr{F} f_b$ for $b = 0.5$.

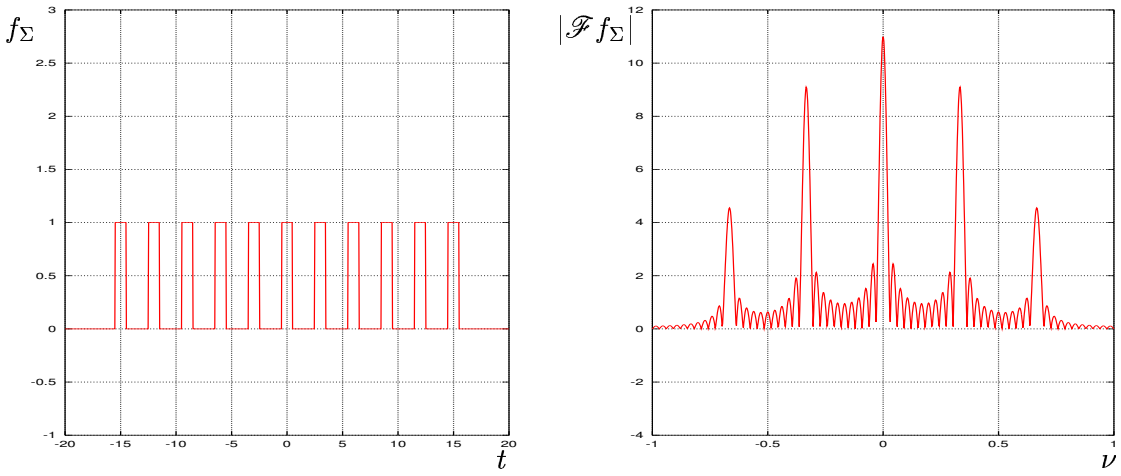


Figure 5.8: The function f_Σ and the absolute of its Fourier transform $|\mathscr{F} f_\Sigma|$ for $b = 0.5$, $a = 3$, $N = 11$.

identical translated copies of f_b. This is the type of time pattern that we would expect from an ideal periodic abstract group. It is depicted in figure 5.8.

From the convolution theorem

$$\mathscr{F}(f_1 \circledast f_2) = \mathscr{F}\left(t \to \int_{-\infty}^{\infty} f_1(t') f_2(t - t') \mathrm{d}t'\right) = (\mathscr{F} f_1)(\mathscr{F} f_2) \tag{5.43}$$

and a convenient expression for functions which are a sum of "repeated" functions of the same form

$$f_\Sigma(t) = (f \circledast \sum_m \delta_{t_m})(t) = \sum_m f(t - t_m) \tag{5.44}$$

where δ_{t_m} is the Dirac delta distribution $< \delta_{t_m}, g(t) >= g(t_m)$, we can easily compute the Fourier transform of such a function f_Σ. Regarding that

$$(\mathscr{F} \delta_{t_m})(\nu) = \mathrm{e}^{-\mathrm{i}2\pi\nu t_m} \tag{5.45}$$

We can compute the Fourier transform of a **finite** and **periodic** sum of N functions f_b, that are b wide and a apart

$$f_\Sigma(t) = (f_b \circledast \sum_{m=-\frac{N-1}{2}}^{\frac{N-1}{2}} \delta_{(m-1)a})(t) = \sum_{m=-\frac{N-1}{2}}^{\frac{N-1}{2}} f_b(t - (m-1)a) \tag{5.46}$$

which is depicted in figure 5.8.
Using (5.46), (5.45) and (5.43) we find that [85]

$$(\mathscr{F} f_\Sigma)(\nu) = (\mathscr{F} f_b)(\nu) \sum_{m=-\frac{N-1}{2}}^{\frac{N-1}{2}} \mathrm{e}^{-\mathrm{i}2\pi\nu(m-1)a} = (\mathscr{F} f_b)(\nu) \frac{\sin(N\pi\nu a)}{\sin(\pi\nu a)} \tag{5.47}$$

This equation is well known from optics where it describes the signal that results from multi-beam interference and diffraction on a grid with slits of finite width. The first term modulates the signal and originates from diffraction, the second term describes the multi-beam interference. If we isolate the second term, we find that as we let the number of slits (the number of Ad-Hoc-Group instantiations) go to infinity, we get [85]

$$\lim_{N\to\infty} \frac{\sin(N\pi\nu a)}{\sin \pi\nu a} = \sum_{m=-\infty}^{\infty} \delta(\nu a - m) \tag{5.48}$$

which is illustrated in figure 5.9.

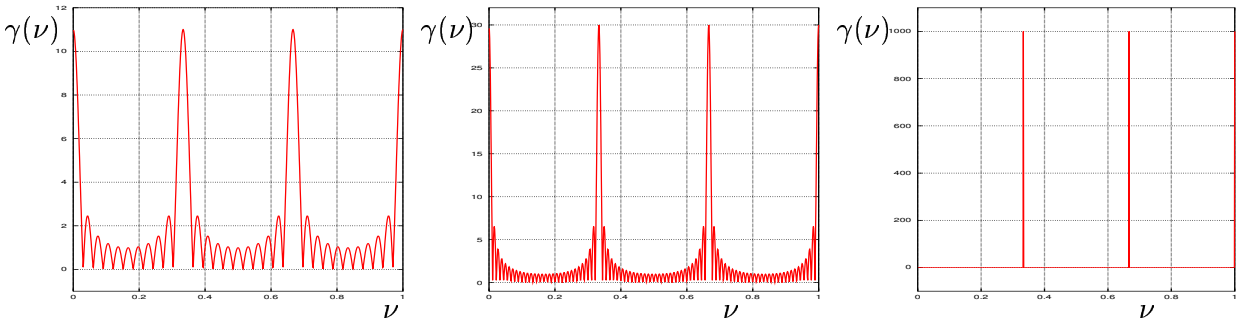


Figure 5.9: The function $\gamma(\nu) = |\frac{\sin(N\pi\nu a)}{\sin \pi\nu a}|$ for $b = 0.5$, $a = 3$ and several values of N: $N = 11$, $N = 30$, $N = 1000$.

What we can see from that is that we just need to determine the second main maximum ($m = 1$) of the Fourier transform, where we can simply **read** the value for the "period" of the abstract group which is

$$a = \frac{1}{\nu_0} \tag{5.49}$$

The position of this second main maximum can be easily determined algorithmically. We made several experiments using the Fast Fourier Transform Algorithm. Although the considerations are relatively stable with respect to minor distortion in the signal from the groups, the results were not very good.
Another procedure which would also yield values for N and b could be to use **fitting techniques** to fit the Fourier transform of the real group signal to the ideal function (5.47). What is appealing in fitting the Fourier transform instead of the original signal is that the Fourier spectrum is comparatively stable with respect to distortions in the signal (in an iteration a group might be wrongly detected or not detected at all). What is also of advantage for the fitting that the Fourier spectrum does not depend on the position of the "window" looking the necessary number of iterations back into the past that is used at a point in time with respect to the group's occurrences. The reason for this is that the Fourier transform's absolute value does not depend on the **phase** φ of the original signal:

$$(\mathscr{F} f_{\Sigma,\varphi})(\nu) = \mathscr{F}\left(f_b \circledast \sum_m \delta_{(m-1)a+\varphi}\right)(\nu) = (\mathscr{F} f_b)(\nu)\, \mathrm{e}^{\mathrm{i}\varphi} \sum_m \mathrm{e}^{-\mathrm{i}2\pi\nu(m-1)a} \tag{5.50}$$

That means, for this detection method of the period of the abstract group it is not of importance when exactly it's Ad-Hoc-Group instantiations meet, but only the time distance of their meetings.

5.1.4.4 Statistical Approach

While theoretically more adapted to a wider range of group signals, the Fourier based method is also quite "bulky" with respect to the computational expenses. A more straightforward approach to analyzing a group's occurrence signal over time is to use a **statistical heuristic** to detect **0-1-transitions** and **1-0 transitions** in the signal corresponding to the beginning and the end of an Ad-Hoc-Group's activity.
Considering a signal of the form depicted in figure 5.6 we use the following simple heuristic to detect a 0-1-transition and a 1-0-transition in the time series f_Σ at position t_k: If in a symmetric time frame $\{t_{k-l}, \ldots, t_{k+l}\}$ around the position t_k the number of zeros to the left of t_k is in $\{l, l-1, \ldots, l-l_{\text{tol}}\}$ and the same is true for the numbers of ones to the right then we assume a 0-1-transition at position t_k. Due to the introduction of a tolerance l_{tol} in the required number of ones and zeros, also **noisy transitions** can be detected. At the same time, the condition may be true for several subsequent positions t_k, but the algorithm only uses the first position for which the condition is fulfilled. The 1-0-detection works analogously.
Formally we have

$$\text{0-1-transition at } t_k = \begin{cases} \text{true} & \text{if } \sum_{m=1}^{l} \left(1 - f_\Sigma(t_{k-m})\right) \geq l - l_{\text{tol}} \ \wedge \ \sum_{m=1}^{l} \left(f_\Sigma(t_{k+m})\right) \geq l - l_{\text{tol}} \\ \text{false} & \text{if } \sum_{m=1}^{l} \left(1 - f_\Sigma(t_{k-m})\right) < l - l_{\text{tol}} \ \vee \ \sum_{m=1}^{l} \left(f_\Sigma(t_{k+m})\right) < l - l_{\text{tol}} \end{cases}$$

$$\text{1-0-transition at } t_k = \begin{cases} \text{true} & \text{if } \sum_{m=1}^{l} \left(1 - f_\Sigma(t_{k+m})\right) \geq l - l_{\text{tol}} \ \wedge \ \sum_{m=1}^{l} \left(f_\Sigma(t_{k-m})\right) \geq l - l_{\text{tol}} \\ \text{false} & \text{if } \sum_{m=1}^{l} \left(1 - f_\Sigma(t_{k+m})\right) < l - l_{\text{tol}} \ \vee \ \sum_{m=1}^{l} \left(f_\Sigma(t_{k-m})\right) < l - l_{\text{tol}} \end{cases}$$

In a signal $f_\Sigma(t_k)$ which we can analyze for s steps ($k \in [0, s]$) we denote the N subsequent positions of the 0-1-transitions by $\{t_{k_1}^{(0-1)}, \ldots, t_{k_N}^{(0-1)}\}$ and the 1-0 transitions analogously. We can average the distances between the 0-1-transitions of Ad-Hoc-Group occurrences and separately for the 1-0-transitions and take the average of both numbers. This will give us a simple, yet effective approximation for the underlying abstract group's period a:

$$a = \frac{1}{2}\left(\frac{1}{N}\sum_{j=1}^{N-1}(t_{k_{j+1}}^{(0-1)} - t_{k_j}^{(0-1)}) + \frac{1}{N}\sum_{j=1}^{N-1}(t_{k_{j+1}}^{(1-0)} - t_{k_j}^{(1-0)})\right) \tag{5.51}$$

Besides the information concerning the period of an Ad-Hoc-Group's probable abstract group, the statistical approach allows us to also determine the average duration b of the Ad-Hoc-Group-Manifestations:

$$b = \frac{1}{N}\sum_{j=1}^{N}(t_{k_j}^{(1-0)} - t_{k_j}^{(0-1)}) \tag{5.52}$$

5.1.4.5 Step 3: Amalgamating Ad-Hoc-Group Models into Abstract Group Models

So what we have mainly **collected so far** is the following information:

- For each run of the detection algorithm (corresponding in our testing scenario to an iteration step of SUMI) the **number of Ad-Hoc-Groups** and **members of each Ad-Hoc-Group**
- For a given number of points in time (runs of the detection algorithm) the **similarity between all Ad-Hoc-Groups** that have been found during this period
- For each of these Ad-Hoc-Groups: **individual occurrence interval length** and (trivially won through the locations and velocities of its members) the **location and velocity** of the group
- For each of these Ad-Hoc-Groups: the **period**, **number of instantiations**, and **average occurrence interval length** of its probable underlying abstract group

This information can be amalgamated into a set of models of the abstract groups which might be slightly different from the probable underlying abstract groups that have been used to compute the periods etc.. The idea is to compute **scores** from the above set of information which decide about the final proposed abstract group structure.
The heuristics used to arrange the abstract groups is based on the following assumptions:

1. The larger the **number of members** an Ad-Hoc-Group is, the more significant it is.
2. The longer its **individual occurrence interval length** is, the more significant it is.
3. The more **instantiations** it has or the more **instantiations** its probable underlying abstract group has, the more significant it is.

The **first two assumptions** are generally not directly backed by socio-psychology. These assumptions arise from considerations of group detection: We use these two assumptions because a large group has to fulfill more conditions to be accepted as a group than a small group (e.g. for every member the condition that it needs to be near enough to the other members). This makes the probability of an accidental detection smaller with growing member count. A similar consideration leads to the second assumption. The longer an Ad-Hoc Group persists the more reliable is its detection assumed to be. The **third assumption** is motivated by common sense as well as social psychology (see chapter 2). It allows us to distinguish Ad-Hoc-Groups that are instantiations of an abstract group from pure Ad-Hoc-Groups which exist e.g. only once. In order to re-compute the abstract groups from the set of Ad-Hoc-Groups, a relevance score for each Ad-Hoc-Group in a row of similar Ad-Hoc-Groups that we have used to determine the period of an Ad-Hoc-Group is computed. The abstract group (row) is then rebuilt from the Ad-Hoc-Group with the maximal score. We set as the period of this final abstract group a weighted sum of the periods [3] of its constituting Ad-Hoc-Groups. The weights are combinations of the pre-score and the similarity of the constituting abstract group and the group with the highest pre-score.

[3] Remember: The period of an Ad-Hoc Group is computed with its **probable abstract group**

Computing Abstract Groups

- Input:
  - For every run of the Ad-Hoc-Group detection Algorithm at time t_k the set of Ad-Hoc-Groups $\{\mathcal{G}_{i_1}^{(t_k)}, \mathcal{G}_{i_2}^{(t_k)}, \dots\}$ which have been detected in this run.
  - For each group $\mathcal{G}_i$ the members of this group $\{k_1(i_1), k_2(i_1), \dots, k_{|\mathcal{G}_i|}(i_1)\}$ which uniquely identify the group. $\{k_1(i_1), k_2(i_1), \dots, k_{|\mathcal{G}_i|}(i_1)\} \stackrel{\text{Hashfct}}{\longrightarrow} i$.
  - Result of Step 1 (see section 5.1.4.1): Similarity matrix $s_{i_1,i_2} = \mathtt{sim}(\mathcal{G}_{i_1}, \mathcal{G}_{i_2})$ with respect to members between all Ad-Hoc-Groups of all runs t_k. Threshold value p for required similarity
  - Result of Step 2 (see section 5.1.4.2): For each group $\mathcal{G}_i$ the period a_i and the average duration b_i of its probable abstract group $\mathcal{A}_i$ (see section 5.1.4.2).
- Process:
  - For (group $i := 0$; $i \leq i_{\max}$; i++)
    - Compute for every Ad-Hoc-Group in the probable abstract group $\mathcal{A}_i = \{\mathcal{G}_{i_m} | \mathtt{sim}(\mathcal{G}_{i_m}, \mathcal{G}_i) \geq p\}$ a prescore η_{i_m}

$$\eta_{i_m} = \frac{1}{3}\left(\frac{|\mathcal{A}_{i_m}|}{\max_n |\mathcal{A}_{i_n}|} + \frac{b_{i_m}}{\max_n b_{i_n}} + \frac{|\mathcal{G}_{i_m}|}{\max_n |\mathcal{G}_{i_n}|}\right) \tag{5.53}$$

    - From the Ad-Hoc-Group with the maximum pre-score $\mathcal{G}_{i_{m'}} : \eta_{i_{m'}} = \max_m \eta_{i_m}$ set as the new abstract group for group $\mathcal{G}_i$:

$$\tilde{\mathcal{A}}_i = \mathcal{A}_{i_{m'}} = \{\mathcal{G}_{i_n} | \mathtt{sim}(\mathcal{G}_{i_n}, \mathcal{G}_{i_{m'}}) \geq p\}$$

    - Compute weighted period for $\tilde{\mathcal{A}}_i$:

$$\tilde{a}_i = \sum_{\{n | \mathcal{G}_{i_n} \in \tilde{\mathcal{A}}_i\}} \frac{1}{2}(\eta_{i_n} + \mathtt{sim}(\mathcal{G}_{i_n}, \mathcal{G}_{i_{m'}}) a_{i_n} \tag{5.54}$$

    Endfor
  - Remove duplicates in the set of newly computed $\tilde{\mathcal{A}}_i$

Figure 5.10: Computing Abstract Groups

5.1.4.6 Comparing Found Abstract Groups with Actual Abstract Groups

From our SUMI simulation we can easily extract the **actual abstract groups**. Each abstract group corresponds to two rows in the group schedule and is instantiated in several actual Ad-

Hoc-Groups. This was already discussed in section 3.8. We also can easily compute the values of a_i and b_i for the actual abstract groups with the help of equations (5.51) and (5.51). Instead of taking the average between the differences of the 0-1-transition difference based value of a_i and the 1-0-transition difference based value of a_i, we only use the 1-0-transition difference based values to calculate a_i because of the characteristics of our simulation: The end of a group period is equal for all its members while at the beginning of a group's interaction period, the members need to move to the meeting point and thus parts of the group may arrive earlier than others. In order to compare the **actual abstract groups** $\mathcal{A}_i^{\text{actual}}$ with the **found abstract groups** $\tilde{\mathcal{A}}_i^{\text{found}}$ that have been computed using the procedure of figure 5.10 we use the method discussed in section 5.1.3.4:

First we compute the similarity $s_{i_1 i_2}^{\text{Members}} = \text{sim}^{\text{Members}}(\mathcal{A}_{i_1}^{\text{actual}}, \tilde{\mathcal{A}}_{i_2}^{\text{found}})$ with respect to the overall members. Here we use the $F_{0.5}$-Measure (equation (5.36)). The set of overall members of an actual abstract group $\{\mathcal{A}_i^{\text{actual}}\} = \{\mathcal{G}_{i_1}^{\text{actual}}, \mathcal{G}_{i_2}^{\text{actual}}, \dots\}$ is computed as the union of the members of its constituing abstract groups $\mathcal{G}_i^{\text{actual}}$. The set of overall members of a found abstract group is computed analogously.
Then we compute the similarity $s_{i_1 i_2}^{\text{Period}}$ with respect to the periods of the abstract groups

$$s_{i_1 i_2}^{\text{Period}} = \text{sim}^{\text{Period}}(\mathcal{A}_{i_1}^{\text{actual}}, \tilde{\mathcal{A}}_{i_2}^{\text{found}}) = \text{sim}^{\text{Period}}(a_{i_1}^{\text{actual}}, \tilde{a}_{i_2}^{\text{found}}) = 1 - \frac{|a_{i_1}^{\text{actual}} - \tilde{a}_{i_2}^{\text{found}}|}{\tilde{a}_{i_2}^{\text{found}}} \quad . \tag{5.55}$$

We **then** combine the two similarities to an overall similarity

$$s_{i_1 i_2}^{\text{Overall}} = s_{i_1 i_2}^{\text{Period}} * s_{i_1 i_2}^{\text{Members}} \tag{5.56}$$

and modify $s_{i_1 i_2}^{\text{Overall}}$ according to equations (5.25) and (5.26):

$$s_{i_1 i_2}^{\text{Overall}} \xrightarrow{(5.25) \text{ and } (5.26)} \bar{\bar{s}}_{i_1 i_2}^{\text{Overall}} \tag{5.57}$$

This modified overall similarity can now be used to compute the **Precision** and **Recall** and the **$F_{0.5}$-Measure** via equations (5.18) - (5.22).

		SUMI Simulation				
		1	2	3	4	5
$\bar{\bar{s}}_{i_1 i_2}^{\text{Overall}}$	Precision P	0.511	0.131	0.469	0.281	0.346
	Recall R	0.562	0.340	0.516	0.478	0.484
	F-Measure $F_{0.5}$	0.535	0.189	0.492	0.354	0.403
$\bar{\bar{s}}_{i_1 i_2}^{\text{Members}}$	Precision P	0.894	0.370	0.896	0.578	0.687
	Recall R	0.983	0.961	0.986	0.983	0.962
	F-Measure $F_{0.5}$	0.936	0.534	0.939	0.728	0.802

Table 5.4: Comparison between period-modulated and pure membership-based quality measures of the abstract group detection algorithm. For step 1 of the abstract group detection algorithm the measure of equation (5.37) with $w_1 = 0.7$ and $w_2 = 0.3$ was used. We furthermore used the statistical approach for the computation of the periods.

The **results** of **quantitatively evaluating** the most important aspects of the process of finding abstract groups from the Ad-Hoc-Groups which was discussed in the previous pages are shown in tables 5.4, 5.5, 5.6 and 5.7.

SUMI Simulation	Actual Abstract Groups		Found Abstract Groups		
	Overall Members	Periods	Overall Members	Periods	Average Score
1	$\mathcal{G}_{i_1} = \{15, 13, 9, 11, 14, 10, 12\}$ $\mathcal{G}_{i_2} = \{23, 21, 20, 25, 24, 22\}$ $\mathcal{G}_{i_3} = \{45, 47, 48, 46\}$ $\mathcal{G}_{i_4} = \{45, 41, 40, 43, 46, 42, 44\}$ $\mathcal{G}_{i_5} = \{34, 32, 36, 30, 33, 31, 35\}$ $\mathcal{G}_{i_6} = \{4, 9, 8, 6, 10, 7, 5\}$ $\mathcal{G}_{i_7} = \{17, 15, 19, 16, 18, 20\}$ $\mathcal{G}_{i_8} = \{36, 40, 38, 37, 39\}$ $\mathcal{G}_{i_9} = \{30, 26, 28, 29, 27\}$ $\mathcal{G}_{i_{10}} = \{2, 4, 1, 3, 5, 0\}$	$a_{i_1} = 400.0$ $a_{i_2} = 419.0$ $a_{i_3} = 403.0$ $a_{i_4} = 403.0$ $a_{i_5} = 424.5$ $a_{i_6} = 422.0$ $a_{i_7} = 380.0$ $a_{i_8} = 394.0$ $a_{i_9} = 399.0$ $a_{i_{10}} = 405.0$	$\mathcal{G}_{i_1} = \{17, 15, 19, 23, 21, 16, 18, 20, 25, 24, 22\}$ $\mathcal{G}_{i_2} = \{15, 13, 8, 9, 11, 14, 7, 10, 12\}$ $\mathcal{G}_{i_3} = \{34, 32, 30, 36, 33, 3135\}$ $\mathcal{G}_{i_4} = \{41, 40, 43, 48, 46, 4244\}$ $\mathcal{G}_{i_5} = \{19, 23, 21, 18, 20, 242522\}$ $\mathcal{G}_{i_6} = \{36, 40, 38, 37, 39\}$ $\mathcal{G}_{i_7} = \{30, 26, 28, 29, 27\}$ $\mathcal{G}_{i_8} = \{45, 47, 41, 40, 43, 48, 46, 42, 44\}$ $\mathcal{G}_{i_9} = \{34, 32, 30, 36, 38, 33, 31, 35, 37\}$ $\mathcal{G}_{i_{10}} = \{2, 4, 1, 3, 5, 0\}$ $\mathcal{G}_{i_{11}} = \{15, 4, 13, 8, 9, 6, 11, 14, 7, 10, 5, 12\}$ $\mathcal{G}_{i_{12}} = \{17, 15, 19, 23, 21, 16, 18, 20, 24, 22\}$ $\mathcal{G}_{i_{13}} = \{45, 47, 48, 46\}$ $\mathcal{G}_{i_{14}} = \{2, 4, 8, 911, 6, 3, 10, 7, 5, 12, 0\}$ $\mathcal{G}_{i_{15}} = \{17, 15, 19, 23, 21, 16, 18, 20, 3\}$ $\mathcal{G}_{i_{16}} = \{45, 41, 40, 43, 46, 42, 39, 44\}$ $\mathcal{G}_{i_{17}} = \{45, 47, 41, 40, 43, 48, 46, 42, 39, 44\}$	$a_{i_1} = 223.2$ $a_{i_2} = 208.0$ $a_{i_3} = 161.7$ $a_{i_4} = 246.2$ $a_{i_5} = 224.2$ $a_{i_6} = 115.4$ $a_{i_7} = 203.5$ $a_{i_8} = 267.3$ $a_{i_9} = 161.9$ $a_{i_{10}} = 162.5$ $a_{i_{11}} = 155.1$ $a_{i_{12}} = 220.4$ $a_{i_{13}} = 352.3$ $a_{i_{14}} = 106.2$ $a_{i_{15}} = 262.9$ $a_{i_{16}} = 253.3$ $a_{i_{17}} = 260.7$	$\bar{\eta}_{i_1} = 0.60$ $\bar{\eta}_{i_2} = 0.59$ $\bar{\eta}_{i_3} = 0.69$ $\bar{\eta}_{i_4} = 0.61$ $\bar{\eta}_{i_5} = 0.69$ $\bar{\eta}_{i_6} = 0.83$ $\bar{\eta}_{i_7} = 0.95$ $\bar{\eta}_{i_8} = 0.66$ $\bar{\eta}_{i_9} = 0.75$ $\bar{\eta}_{i_{10}} = 0.77$ $\bar{\eta}_{i_{11}} = 0.67$ $\bar{\eta}_{i_{12}} = 0.65$ $\bar{\eta}_{i_{13}} = 0.74$ $\bar{\eta}_{i_{14}} = 0.62$ $\bar{\eta}_{i_{15}} = 0.71$ $\bar{\eta}_{i_{16}} = 0.76$ $\bar{\eta}_{i_{17}} = 0.66$
2	$\mathcal{G}_{i_1} = \{15, 13, 11, 14, 10, 12\}$ $\mathcal{G}_{i_2} = \{19, 23, 21, 20, 25, 24, 22\}$ $\mathcal{G}_{i_3} = \{45, 47, 48, 46\}$ $\mathcal{G}_{i_4} = \{45, 41, 40, 43, 42, 44\}$ $\mathcal{G}_{i_5} = \{34, 32, 30, 33, 31, 35\}$ $\mathcal{G}_{i_6} = \{8, 6, 7, 5\}$ $\mathcal{G}_{i_7} = \{17, 15, 19, 16, 18, 20\}$ $\mathcal{G}_{i_8} = \{34, 36, 38, 35, 37, 39\}$ $\mathcal{G}_{i_9} = \{30, 26, 28, 29, 27\}$ $\mathcal{G}_{i_{10}} = \{2, 4, 1, 3, 5\}$	$a_{i_1} = 410.5$ $a_{i_2} = 401.0$ $a_{i_3} = 392.0$ $a_{i_4} = 405.0$ $a_{i_5} = 401.5$ $a_{i_6} = 435.0$ $a_{i_7} = 424.0$ $a_{i_8} = 425.0$ $a_{i_9} = 418.0$ $a_{i_{10}} = 422.5$	$\mathcal{G}_{i_1} = \{30, 26, 28, 29, 27\}$ $\mathcal{G}_{i_2} = \{34, 32, 30, 38, 33, 31, 35, 37, 39\}$ $\mathcal{G}_{i_3} = \{45, 47, 43, 48, 46, 44\}$ $\mathcal{G}_{i_4} = \{2, 4, 9, 8, 6, 1, 3, 7, 5\}$ $\mathcal{G}_{i_5} = \{45, 47, 41, 40, 43, 48, 46, 42, 44\}$ $\mathcal{G}_{i_6} = \{15, 13, 9, 11, 14, 10, 12\}$ $\mathcal{G}_{i_7} = \{17, 19, 23, 21, 16, 18, 20, 25, 24, 22\}$ $\mathcal{G}_{i_8} = \{17, 15, 19, 16, 18, 20\}$ $\mathcal{G}_{i_9} = \{34, 36, 38, 35, 37, 39\}$ $\mathcal{G}_{i_{10}} = \{17, 15, 19, 21, 16, 18, 20, 24, 22\}$ $\mathcal{G}_{i_{11}} = \{45, 47, 41, 40, 43, 46, 42, 44\}$ $\mathcal{G}_{i_{12}} = \{19, 23, 21, 20, 25, 24, 22\}$	$a_{i_1} = 192.8$ $a_{i_2} = 307.7$ $a_{i_3} = 250.3$ $a_{i_4} = 202.0$ $a_{i_5} = 169.4$ $a_{i_6} = 112.3$ $a_{i_7} = 203.1$ $a_{i_8} = 387.9$ $a_{i_9} = 380.4$ $a_{i_{10}} = 343.0$ $a_{i_{11}} = 137.4$ $a_{i_{12}} = 147.9$	$\bar{\eta}_{i_1} = 0.74$ $\bar{\eta}_{i_2} = 0.68$ $\bar{\eta}_{i_3} = 0.75$ $\bar{\eta}_{i_4} = 0.65$ $\bar{\eta}_{i_5} = 0.61$ $\bar{\eta}_{i_6} = 0.68$ $\bar{\eta}_{i_7} = 0.79$ $\bar{\eta}_{i_8} = 0.66$ $\bar{\eta}_{i_9} = 0.77$ $\bar{\eta}_{i_{10}} = 0.80$ $\bar{\eta}_{i_{11}} = 0.63$ $\bar{\eta}_{i_{12}} = 0.61$

Table 5.5: Overall Members, periods and average Score of the Abstract Actual and Found Groups resulting from the procedure shown in figure 5.10. The average score for an Abstract Found Group $\tilde{\mathcal{A}}_i = \{\mathcal{G}_{i_n} | \text{sim}(\mathcal{G}_{i_n}, \mathcal{G}_{i_{m'}}) \geq p\}$ is computed as $\bar{\eta}_i = \frac{1}{|\tilde{\mathcal{A}}_i|} \sum_{\{n | \mathcal{G}_{i_n} \in \tilde{\mathcal{A}}_i\}} \frac{1}{2}(\eta_{i_n} + \text{sim}(\mathcal{G}_{i_n}, \mathcal{G}_{i_{m'}}))$ (Refer to figure 5.10 for the precise index semantics). For step 1 of the abstract group detection algorithm the measure of equation (5.37) with the sub-optiomal setting $w_1 = 0.3$ and $w_2 = 0.7$ was used. We furthermore used the statistical approach for the computation of the periods.

Table 5.4 shows the **overall results** of the process for five different SUMI simulations. The upper row shows the results in form of Precision, Recall and F-Measure computed with the help of the overall similarity measure of equations (5.56), (5.57) which modulates the membership based similarity with the similarity with respect to the periods. The second row shows values computed with the membership based similarity alone.
What we can see is that the performance of the abstract group detection procedure is **very good** with respect to the **members** of the abstract group. The performance of the **period detection** is **less satisfactory** which we can also see from tables 5.5 and 5.6. The periods detected are about 50 per cent to small which yields values for Precision, Recall and F-Measure which are also about half as good for the period-modulated similarity (compare equations (5.55) and (5.56)).

Tables 5.5 and 5.6 illustrate the results with respect to **overall members** and **periods** for a suboptimal setting of the step 1 parameters w_1 and w_2 which is better suited to show the possible errors in the results. What we can see is that the actual abstract groups from the simulation are well reflected in the found abstract groups (remember that a better setting of the w_1 and w_2 will deliver a better congruence (compare table 5.7)). The rightmost column

SUMI Simulation	Actual Abstract Groups		Found Abstract Groups		
	Overall Members	Periods	Overall Members	Periods	Average Score
3	$\mathcal{G}_{i_1} = \{15, 13, 9, 11, 14, 10, 12\}$ $\mathcal{G}_{i_2} = \{26, 23, 21, 25, 24, 22\}$ $\mathcal{G}_{i_3} = \{45, 47, 48, 46\}$ $\mathcal{G}_{i_4} = \{45, 41, 43, 46, 42, 44\}$ $\mathcal{G}_{i_5} = \{34, 32, 30, 33, 31, 35\}$ $\mathcal{G}_{i_6} = \{9, 8, 11, 6, 10, 7\}$ $\mathcal{G}_{i_7} = \{17, 15, 19, 16, 18, 14\}$ $\mathcal{G}_{i_8} = \{34, 36, 40, 38, 35, 37, 39\}$ $\mathcal{G}_{i_9} = \{30, 26, 28, 31, 29, 27\}$ $\mathcal{G}_{i_{10}} = \{2, 4, 1, 3, 5, 0\}$	$a_{i_1} = 401.0$ $a_{i_2} = 424.5$ $a_{i_3} = 426.0$ $a_{i_4} = 415.0$ $a_{i_5} = 417.5$ $a_{i_6} = 412.0$ $a_{i_7} = 410.0$ $a_{i_8} = 411.0$ $a_{i_9} = 390.0$ $a_{i_{10}} = 415.0$	$\mathcal{G}_{i_1} = \{32, 34, 36, 30, 38, 33, 31, 29, 35, 37\}$ $\mathcal{G}_{i_2} = \{13, 8, 9, 6, 11, 14, 7, 10, 12\}$ $\mathcal{G}_{i_3} = \{32, 30, 26, 33, 28, 31, 29, 27\}$ $\mathcal{G}_{i_4} = \{32, 34, 30, 26, 28, 33, 31, 29, 35, 27\}$ $\mathcal{G}_{i_5} = \{45, 47, 41, 43, 48, 46, 42, 44\}$ $\mathcal{G}_{i_6} = \{26, 23, 21, 25, 24, 22\}$ $\mathcal{G}_{i_7} = \{17, 15, 19, 16, 18, 20, 14\}$ $\mathcal{G}_{i_8} = \{32, 34, 36, 30, 40, 38, 33, 31, 35, 37, 39\}$ $\mathcal{G}_{i_9} = \{15, 13, 8, 9, 6, 11, 14, 7, 10, 12\}$ $\mathcal{G}_{i_{10}} = \{2, 4, 1, 3, 5, 0\}$ $\mathcal{G}_{i_{11}} = \{34, 36, 40, 38, 35, 37, 39\}$	$a_{i_1} = 199.2$ $a_{i_2} = 266.9$ $a_{i_3} = 157.9$ $a_{i_4} = 165.4$ $a_{i_5} = 200.8$ $a_{i_6} = 208.0$ $a_{i_7} = 149.1$ $a_{i_8} = 259.1$ $a_{i_9} = 248.4$ $a_{i_{10}} = 220.7$ $a_{i_{11}} = 324.5$	$\bar{\eta}_{i_1} = 0.67$ $\bar{\eta}_{i_2} = 0.61$ $\bar{\eta}_{i_3} = 0.64$ $\bar{\eta}_{i_4} = 0.65$ $\bar{\eta}_{i_5} = 0.79$ $\bar{\eta}_{i_6} = 0.74$ $\bar{\eta}_{i_7} = 0.66$ $\bar{\eta}_{i_8} = 0.68$ $\bar{\eta}_{i_9} = 0.67$ $\bar{\eta}_{i_{10}} = 0.73$ $\bar{\eta}_{i_{11}} = 0.66$
4	$\mathcal{G}_{i_1} = \{13, 9, 11, 14, 10, 12\}$ $\mathcal{G}_{i_2} = \{23, 21, 20, 25, 24, 22\}$ $\mathcal{G}_{i_3} = \{45, 47, 48, 46\}$ $\mathcal{G}_{i_4} = \{41, 40, 43, 42, 44\}$ $\mathcal{G}_{i_5} = \{34, 32, 30, 33, 31, 35\}$ $\mathcal{G}_{i_6} = \{9, 8, 6, 10, 7\}$ $\mathcal{G}_{i_7} = \{17, 19, 16, 18, 20\}$ $\mathcal{G}_{i_8} = \{36, 40, 38, 35, 37, 39\}$ $\mathcal{G}_{i_9} = \{30, 26, 28, 29, 25, 27\}$ $\mathcal{G}_{i_{10}} = \{2, 4, 1, 3, 0\}$	$a_{i_1} = 408.5$ $a_{i_2} = 405.5$ $a_{i_3} = 419.0$ $a_{i_4} = 420.5$ $a_{i_5} = 416.5$ $a_{i_6} = 399.0$ $a_{i_7} = 431.0$ $a_{i_8} = 422.0$ $a_{i_9} = 390.0$ $a_{i_{10}} = 416.0$	$\mathcal{G}_{i_1} = \{36, 40, 38, 35, 37, 39\}$ $\mathcal{G}_{i_2} = \{30, 23, 26, 28, 29, 24, 25, 27\}$ $\mathcal{G}_{i_3} = \{2, 4, 1, 3, 0\}$ $\mathcal{G}_{i_4} = \{41, 30, 36, 40, 38, 33, 18, 35, 42, 37, 39, 44\}$ $\mathcal{G}_{i_5} = \{45, 47, 48, 46, 44\}$ $\mathcal{G}_{i_6} = \{13, 8, 9, 11, 6, 7, 10, 12\}$ $\mathcal{G}_{i_7} = \{34, 32, 30, 33, 31, 35\}$ $\mathcal{G}_{i_8} = \{41, 40, 43, 42, 44\}$ $\mathcal{G}_{i_9} = \{13, 8, 9, 6, 11, 14, 7, 10, 12\}$ $\mathcal{G}_{i_{10}} = \{2, 4, 43, 1, 3, 0\}$ $\mathcal{G}_{i_{11}} = \{26, 23, 21, 20, 25, 24, 27, 22\}$ $\mathcal{G}_{i_{12}} = \{30, 26, 23, 28, 21, 20, 29, 25, 24, 27, 22\}$ $\mathcal{G}_{i_{13}} = \{17, 19, 16, 18, 20\}$ $\mathcal{G}_{i_{14}} = \{41, 36, 40, 38, 35, 37, 42, 39, 44\}$ $\mathcal{G}_{i_{15}} = \{30, 36, 40, 38, 33, 18, 35, 37, 39\}$	$a_{i_1} = 301.4$ $a_{i_2} = 202.8$ $a_{i_3} = 404.3$ $a_{i_4} = 251.5$ $a_{i_5} = 168.2$ $a_{i_6} = 233.9$ $a_{i_7} = 116.6$ $a_{i_8} = 202.1$ $a_{i_9} = 175.4$ $a_{i_{10}} = 408.7$ $a_{i_{11}} = 150.3$ $a_{i_{12}} = 163.7$ $a_{i_{13}} = 121.2$ $a_{i_{14}} = 276.2$ $a_{i_{15}} = 273.0$	$\bar{\eta}_{i_1} = 0.65$ $\bar{\eta}_{i_2} = 0.66$ $\bar{\eta}_{i_3} = 0.74$ $\bar{\eta}_{i_4} = 0.64$ $\bar{\eta}_{i_5} = 0.84$ $\bar{\eta}_{i_6} = 0.65$ $\bar{\eta}_{i_7} = 0.54$ $\bar{\eta}_{i_8} = 0.78$ $\bar{\eta}_{i_9} = 0.67$ $\bar{\eta}_{i_{10}} = 0.74$ $\bar{\eta}_{i_{11}} = 0.64$ $\bar{\eta}_{i_{12}} = 0.72$ $\bar{\eta}_{i_{13}} = 0.89$ $\bar{\eta}_{i_{14}} = 0.69$ $\bar{\eta}_{i_{15}} = 0.74$
5	$\mathcal{G}_{i_1} = \{13, 9, 11, 14, 10, 12\}$ $\mathcal{G}_{i_2} = \{23, 21, 20, 25, 24, 22\}$ $\mathcal{G}_{i_3} = \{45, 47, 48, 46\}$ $\mathcal{G}_{i_4} = \{45, 41, 40, 43, 46, 42, 44\}$ $\mathcal{G}_{i_5} = \{34, 32, 30, 33, 31, 35\}$ $\mathcal{G}_{i_6} = \{9, 8, 6, 10, 7\}$ $\mathcal{G}_{i_7} = \{17, 15, 19, 16, 18, 20\}$ $\mathcal{G}_{i_8} = \{41, 36, 40, 38, 37, 39\}$ $\mathcal{G}_{i_9} = \{30, 26, 28, 29, 25, 27\}$ $\mathcal{G}_{i_{10}} = \{2, 4, 1, 3, 0\}$	$a_{i_1} = 396.0$ $a_{i_2} = 403.0$ $a_{i_3} = 404.0$ $a_{i_4} = 425.0$ $a_{i_5} = 420.0$ $a_{i_6} = 398.0$ $a_{i_7} = 413.0$ $a_{i_8} = 440.0$ $a_{i_9} = 413.0$ $a_{i_{10}} = 412.0$	$\mathcal{G}_{i_1} = \{45, 47, 41, 26, 40, 43, 48, 46, 42, 44\}$ $\mathcal{G}_{i_2} = \{17, 19, 23, 21, 18, 20, 25, 24, 22\}$ $\mathcal{G}_{i_3} = \{30, 26, 28, 29, 24, 25, 27\}$ $\mathcal{G}_{i_4} = \{34, 32, 30, 33, 31, 35\}$ $\mathcal{G}_{i_5} = \{13, 9, 11, 14, 10, 12\}$ $\mathcal{G}_{i_6} = \{23, 21, 20, 25, 24, 22\}$ $\mathcal{G}_{i_7} = \{45, 41, 36, 40, 38, 43, 37, 39, 44\}$ $\mathcal{G}_{i_8} = \{45, 47, 26, 43, 48, 46, 42, 44\}$ $\mathcal{G}_{i_9} = \{17, 19, 23, 21, 18, 20, 24, 22\}$ $\mathcal{G}_{i_{10}} = \{17, 15, 19, 16, 18, 20\}$ $\mathcal{G}_{i_{11}} = \{26, 23, 21, 20, 25, 24, 27, 22\}$ $\mathcal{G}_{i_{12}} = \{4, 9, 8, 6, 10, 7, 5\}$ $\mathcal{G}_{i_{13}} = \{2, 4, 1, 3, 0\}$ $\mathcal{G}_{i_{14}} = \{45, 47, 41, 40, 38, 43, 46, 37, 42, 39, 44\}$ $\mathcal{G}_{i_{15}} = \{4, 9, 8, 11, 6, 14, 10, 7, 12, 5\}$ $\mathcal{G}_{i_{16}} = \{45, 47, 41, 40, 38, 43, 48, 46, 42, 44\}$ $\mathcal{G}_{i_{17}} = \{13, 8, 9, 6, 11, 14, 7, 10, 12\}$ $\mathcal{G}_{i_{18}} = \{13, 9, 11, 6, 14, 10, 12\}$ $\mathcal{G}_{i_{19}} = \{41, 36, 40, 38, 37, 39\}$	$a_{i_1} = 207.6$ $a_{i_2} = 309.0$ $a_{i_3} = 238.5$ $a_{i_4} = 205.4$ $a_{i_5} = 320.3$ $a_{i_6} = 320.2$ $a_{i_7} = 232.9$ $a_{i_8} = 199.2$ $a_{i_9} = 309.8$ $a_{i_{10}} = 176.4$ $a_{i_{11}} = 296.1$ $a_{i_{12}} = 199.5$ $a_{i_{13}} = 271.0$ $a_{i_{14}} = 192.6$ $a_{i_{15}} = 199.1$ $a_{i_{16}} = 196.6$ $a_{i_{17}} = 291.8$ $a_{i_{18}} = 297.7$ $a_{i_{19}} = 258.9$	$\bar{\eta}_{i_1} = 0.62$ $\bar{\eta}_{i_2} = 0.69$ $\bar{\eta}_{i_3} = 0.66$ $\bar{\eta}_{i_4} = 0.81$ $\bar{\eta}_{i_5} = 0.63$ $\bar{\eta}_{i_6} = 0.69$ $\bar{\eta}_{i_7} = 0.67$ $\bar{\eta}_{i_8} = 0.72$ $\bar{\eta}_{i_9} = 0.68$ $\bar{\eta}_{i_{10}} = 0.75$ $\bar{\eta}_{i_{11}} = 0.71$ $\bar{\eta}_{i_{12}} = 0.73$ $\bar{\eta}_{i_{13}} = 0.84$ $\bar{\eta}_{i_{14}} = 0.71$ $\bar{\eta}_{i_{15}} = 0.66$ $\bar{\eta}_{i_{16}} = 0.60$ $\bar{\eta}_{i_{17}} = 0.74$ $\bar{\eta}_{i_{18}} = 0.69$ $\bar{\eta}_{i_{19}} = 0.70$

Table 5.6: Overall Members, periods and average Score of the Abstract Actual and Found Groups resulting from the procedure shown in figure 5.10. The average score for an Abstract Found Group $\tilde{\mathcal{A}}_i = \{\mathcal{G}_{i_n} | \text{sim}(\mathcal{G}_{i_n}, \mathcal{G}_{i_{m'}}) \geq p\}$ is computed as $\bar{\eta}_i = \frac{1}{|\tilde{\mathcal{A}}_i|} \sum_{\{n|\mathcal{G}_{i_n} \in \tilde{\mathcal{A}}_i\}} \frac{1}{2}(\eta_{i_n} + \text{sim}(\mathcal{G}_{i_n}, \mathcal{G}_{i_{m'}}))$ (Refer to figure 5.10 for the precise index semantics). For step 1 of the abstract group detection algorithm the measure of equation (5.37) with the sub-optiomal setting $w_1 = 0.3$ and $w_2 = 0.7$ was used. We furthermore used the statistical approach for the computation of the periods.

shows the average score of the Ad-Hoc-Groups that have contributed to the particular abstract group. Such an average score for an Abstract Found Group $\tilde{\mathcal{A}}_i = \{\mathcal{G}_{i_n} | \text{sim}(\mathcal{G}_{i_n}, \mathcal{G}_{i_{m'}}) \geq p\}$ is computed as $\bar{\eta}_i = \frac{1}{|\tilde{\mathcal{A}}_i|} \sum_{\{n|\mathcal{G}_{i_n} \in \tilde{\mathcal{A}}_i\}} \frac{1}{2}(\eta_{i_n} + \text{sim}(\mathcal{G}_{i_n}, \mathcal{G}_{i_{m'}}))$ (Refer to figure 5.10 for the precise index semantics). What we can see is that the **average score** is significantly higher for groups with a higher congruence with respect to members with the actual abstract groups. This shows

		Similarity Measure $\text{sim}(\mathcal{G}_{i_1}, \mathcal{G}_{i_2})$ used in Step 1				
		Hamming Distance Based	$F_{0.5}$-Measure	Measure from equation (5.37)		
				$w_1 = 0.3$ $w_2 = 0.7$	$w_1 = 0.5$ $w_2 = 0.5$	$w_1 = 0.7$ $w_2 = 0.3$
Fourier	Precision P	0.226	0.116	0.216	0.232	0.258
	Recall R	0.361	0.163	0.345	0.348	0.335
	F-Measure $F_{0.5}$	0.278	0.136	0.266	0.278	0.291
Statistical	Precision P	0.306	0.128	0.139	0.197	0.335
	Recall R	0.490	0.179	0.223	0.295	0.436
	F-Measure $F_{0.5}$	0.377	0.149	0.171	0.236	0.379

Table 5.7: Comparing the alternatives for the similarity measure for comparing Ad-Hoc-Groups with respect to their members (step 2 of the overall abstract group detection procedure) and the two alternatives for period detection (step 1 of the overall abstract group detection procedure). The Fourier based calculations use the statistical approach to compute the number of instantiations and the average length of the Ad-Hoc-Group intervals which are used in the calculation of the scores (compare figure 5.10).

that the average score is a **valid tool** for quantitatively characterizing the probable quality of a found abstract group. It a posteriori justifies the construction of the score shown in figure 5.10 and the theses that have been stated in 5.1.4.5 that were used as a basis for the score construction.

Tables 5.7 compares two aspects of the abstract group detection algorithm. First the various **symmetric similarity measures** that have been proposed for the comparison of Ad-Hoc-Groups with respect to their members (step 1 of the overall procedure) in section 5.1.4.1. Second, the two approaches for **period detection** (step 2) that have been introduced in sections 5.1.4.3 and 5.1.4.4 are compared. What is apparent from the table is that the statistical approach performs better in most cases because it is the simpler. more robust approach. This has already been discussed in section 5.1.4.4. Surprisingly enough the simple Hamming distance based similarity outperforms the F-measure based similarity significantly. This was unexpected from the form of the measure because the Hamming distance does not incorporate the size of the groups. If we would test the algorithm with a simulation that uses a larger stochastic spread in the size of the groups, we might get less from the Hamming based measure. What performs best is the measure proposed in equation (5.37) if the right weights are used. It works best, when the majority of the similarity comes from Hamming distance but a smaller contribution from the modification that takes into account the number of coinciding members vs. the average lengths of the groups boosts the performance of the combined measure beyond that of pure the Hamming based.

5.2 Detection and Modeling of Groups on the Basis of Interests

We have seen in the previous sections, how profile elements with strong dynamics (**highly contextual parameters**) such as location and velocity can be used to detect Ad-Hoc-Groups and how we can also use them to draw conclusions on the structure of the underlying abstract group structure of a mobile community. As has been discussed in chapters 3 and 4, a user profile contains many more elements which can be used to identify group structures. Among these, a **user's interests** play an important role. In chapter 4 we have intensively discussed, how **sets of free text interest phrases** can be compared with the help of similarity measures and also how **list-of-choice interest vectors** can be compared with the help of similarity measures. We have discussed two main test collections for each case which consist of 100 such sets and 100 such vectors respectively.
The test collection "Survey Collection" of sets of free text interest phrases is shown in figures C.1 and C.2 in the appendix. The test collection "Dating Collection" of list-of-choice interest vectors is shown in figures C.3 and C.4 in the appendix. The similarity measure to compare sets of free text interest phrases is shown in figures 4.9 and 4.10 and the similarity measure to compare list-of-choice interest vectors is shown in figure 4.5. What we get in each case is a 100×100 **matrix of similarities** in $[0, 1]$. Parts of these matrices are shown in figures C.7 and C.8. The whole matrix with 10^4 entries was not shown due to space restrictions.
These two matrices are our starting point in investigating how we can use these similarity measures to detect group structures in the interests of the users. The **heuristic** on which the process will be built on, is that the interests within a group are expected to be more similar (on average) than interests between users from different groups. We have discussed and justified this heuristic in chapter 2.
The group detection on the basis of locations and velocities could rely on crisp clustering techniques because we were able to formulate clear criteria on what clusters we want to accept as a group. These criteria were based on considerations discussed in chapter 2. Either people are engaged in social interaction in a generalized spatio-temporal situation or they are not. Cases of overlapping groups are not very common. Thus we were able to rely on non-overlapping clusterings and cluster algorithms and amalgamated the resulting non-overlapping Ad-Hoc-Group models into abstract group models in a later step. The abstract groups could then in principle be overlapping. In case of group detection on the basis of interests, the situation is less easy. Overlapping and related interests may also occur between members of different groups. It is often even hard to say were one group ends and the other begins. Overlapping groups are the rule rather than the exception. We will thus have to switch to a **fuzzy set view for groups** and will thus have to investigate the basic principles of **fuzzy clustering algorithms** before we can continue.

5.2.1 Fuzzy Clustering

As we have stated before, fuzzy sets allow to specify a continuous degree of membership of an element $x_k \in \mathcal{X}$ in a set $\mathcal{C}_i$. A fuzzy set $\mathcal{C}_i$ is characterized by its fuzzy membership function $\mu_i : \mathcal{X} \to [0, 1]$. As was discussed before, a fuzzy classifier/clusterer assigns to a pattern a degree of membership in every cluster. The result is a fuzzy partition (clustering, classification) of the set of patterns $\mathcal{X}$. We can view the values of the membership-functions $\mu_i(x_k \in \mathcal{X})$ as a matrix U_{ik}.

5.2.1.1 Fuzzy C-Means

As we have mentioned while discussing the crisp K-Means algorithm, the result of a clustering algorithm can be represented in different forms. The results can e.g. be represented in form of a partition matrix U or with the help of cluster prototypes. A partition-matrix element U_{ik} represents the degree of pattern x_k belonging to cluster $\mathcal{C}_i$. The degrees can be either crisp $U_{ik} \in \{0, 1\}$ or fuzzy $U_{ik} \in [0, 1]$.
Cluster prototypes aim at representing the cluster by an element of the universe under investigation which is most typical for the cluster. In case of a universe $\mathcal{X} = \mathbb{R}^m$, prototypes of clusters $\mathcal{C}_i$ would be real m-dimensional vectors π_i and we can generate a crisp non-overlapping partition of the set $\mathcal{X}$ into c clusters by applying the nearest-neighbor rule (equation 5.5).
As was discussed in section 5.1.2.2 we can use optimization strategies to compute cluster prototypes. The functions J to be optimized can be chosen according to many paradigms [77] [161]. One of the simplest functions is the square error criterion that we have used in the case of the K-Means algorithm. It is a quantitative measure for the intra-cluster-(un)compactness. Formulated with the membership matrix U_{ik} it reads

$$J_{SQE} = \sum_{k=1}^{n} \sum_{i=1}^{c} U_{ik} ||x_k - \pi_i||^2 \tag{5.58}$$

In case of non-overlapping crisp clustering, the optimization condition $\mathrm{d}J_{SQE}/\mathrm{d}\pi_i = 0$ together with the non-overlapping-constraint $\forall k(\exists i(U_{ik} = 1) \wedge ((j \neq i) \rightarrow (U_{jk} = 0)))$ leads to $\pi_i = \sum_{k=1}^{n} U_{ik} x_k / \sum_{k=1}^{n} U_{ik} = (1/|\mathcal{C}_i|) \sum_{x_k \in \mathcal{C}_i} x_k$ which we know from the discussion of crisp K-Means.
In case of fuzzy clustering we need to modify the criterion function to a generalized square error criterion [161]

$$J_{GSQE} = \sum_{k=1}^{n} \sum_{i=1}^{c} U_{ik}^m ||x_k - \pi_i||^2 \tag{5.59}$$

By introducing the power m one can chose the degree of fuzziness: In the limit $m = 1$ we receive the crisp square error function with no fuzziness. In the limit $m \rightarrow \infty$ we arrive at $\forall i, k(U_{ik} = 1/c)$ that is every element of $\mathcal{X}$ has the same degree of belonging to every class (complete fuzziness) which can be seen by the following discussion.
First we need to define two notions which we will need later in the discussion.
A partition is called **probabilistic** if

$$\forall\, x_k \quad : \quad \sum_{i=1}^{c} \mu_i(x_k) = 1 \tag{5.60}$$

$$\forall\, \mathcal{C}_i \quad : \quad \sum_{k=1}^{|\mathcal{X}|} \mu_i(x_k) > 0 \tag{5.61}$$

A partition is called **possibilistic** if the second condition (5.61) holds. Condition (5.60) states that every pattern must belong to each of the classes in a way that the overall sum of the degrees is constant for all patterns. The condition (5.61) means that none of the classes is empty. These conditions allow to view the resulting membership degrees as probability values under certain circumstances [73], which is not necessary for our considerations.

By optimizing J_{GSQE} under the probabilistic constraints $\sum_{i=1}^{c} U_{ik} = 1$ and $\sum_{k=1}^{|\mathcal{X}|} U_{ik} > 0$ with e.g. a Lagrange multiplier technique we get [161][73]

$$U_{ik} = \Big(\sum_{j=1}^{c} \big(\frac{||x_k - \pi_i||}{||x_k - \pi_j||}\big)^{\frac{2}{m-1}}\Big)^{-1} \tag{5.62}$$

$$\pi_i = \frac{\sum_{k=1}^{n} U_{ik}^m x_k}{\sum_{k=1}^{n} U_{ik}} \tag{5.63}$$

This result is obtained assuming that none of the patterns and cluster prototypes coincide:

$$\forall\, i,k: \quad ||x_k - \pi_i|| \neq 0 \tag{5.64}$$

If such cases occur, they can easily be handled by setting values of $U_{ik} = 1$ for the pattern that coincides with the prototype and $U_{ik} = 0$ for all others. If more than one pattern coincides with a cluster's prototype a solution which respects the conditions 5.60 and 5.61 and which respects the intended semantics of the application needs to be defined.
By analyzing (5.62) in the limit $m \to \infty$, one gets

$$U_{ik} \overset{m\to\infty}{\longrightarrow} \frac{1}{\sum_{j=1}^{c} 1} = \frac{1}{c} \tag{5.65}$$

By analyzing (5.62) in the limit $m \to 0$, one arrives at the nearest neighbour rule (5.5). This can be seen by regarding that (5.62) can be rewritten to $U_{ik} = 1/((\sum_{j\neq i}(\frac{||x_k-\pi_i||}{||x_k-\pi_j||})^{\frac{2}{m-1}}) + 1)$, and in this expression the sum in the denominator becomes ∞ in the limit $m \to 1$ when $||x_k - \pi_i|| \neq \min_{1\leq j\leq c} ||x_k - \pi_j||$ and becomes 0 if $||x_k - \pi_i|| = \min_{1\leq j\leq c} ||x_k - \pi_j||$.
As in the case of K-Means Clustering, the computation of the partition matrix U_{ik} depends on the class prototypes π_i and vice versa. It is therefore again necessary to adapt the class prototypes after each computation of the partition matrix U_{ik} or vice versa until the optimization iteration converges (Alternating Optimization) [161]. The precise algorithm is formulated in figure 5.11. In order to determine the optimum number of clusters, a large number of cluster validation strategies are available which allow for selecting the clustering with the optimum number of clusters c. Again, as in the case of the K-Means crisp clustering, we can use the value of the objective function 5.59 as a quality criterion to select c. A large number of other validation measures for fuzzy clusterings exist (see [60] and [61] which we will not discuss here in greater detail).

5.2.1.2 Fuzzy Clustering of Locations

The concept of **Fuzzy Clustering** was also tested for **locations and velocities**. As was mentioned before, the results were not very encouraging. First, it is difficult to apply a simple heuristic in the selection of the resulting clusters. The socially motivated selection process would have to be modified in the case of fuzzy clusters. What is even more disadvantageous is that clusters of the needed quality that would pass such selection tests are not delivered by the fuzzy clustering algorithms which are available. We extensively experimented with moving the social heuristics into the distance measure (remember the discussion on exponentially modulated distance measures in sections 5.1.3.5 and 3.10) but the results did not fulfill the expectations. Furthermore, Fuzzy clusters are less easy to monitor over several runs of the algorithms (over

Fuzzy C-Means Clustering

- Input:
  - Number of clusters c.
  - Set of patterns $\mathcal{X} = \{x_{k_1}, x_{k_2}, \ldots, x_{k_n}\}$
  - Maximum number of iterations $t_{\max}$
  - stopping criterion bound ϵ
  - Suitable metric $||\ ||$
- Initialize:
  - Determine randomly the initial set of prototypes $\Pi^{(0)} = \{\pi_1^{(0)}, \pi_2^{(0)}, \ldots, \pi_c^{(0)}\}$
- Process:
  - For (step $t := 0;\ t \leq t_{\max};\ t$++)
    - Compute $U^{(t)}(\Pi^{(t-1)}, \mathcal{X})$ using equation (5.62) with $m = 2$
    - Compute $\Pi^{(t)}(U^{(t)}, \mathcal{X})$ using equation (5.63) with $m = 2$
    - If $(\forall i\ ||\pi_i^{(t)} - \pi_i^{(t-1)}|| < \epsilon)$ break.
- Output:
  - Partition-Matrix $U \in [0,1]^{c \times n}$ and cluster prototypes $\Pi = \{\pi_1, \pi_2, \ldots, \pi_c\}$.

Figure 5.11: Algorithm for Fuzzy C-Means Clustering (adapted from [161])

time). While we can easily characterize a cluster by a row in the membership matrix U_{ij}, it is less easy to say to which clusters the rows correspond in the next run of the clustering algorithm. Thus it is harder to map a cluster to a unique index i over several runs. In the thesis [37], the stability behavior of fuzzy clustering algorithms was extensively investigated. By stability we mean how the mapping between columns of U_{ij} and the clusters changes when the patterns are gradually changed. Figure 5.12 shows the results of applying the Fuzzy C-Means algorithm to a SUMI simulation. We used the Partition Coefficient (see [37]) as a cluster validation measure to compute the optimum number of clusters. The figure shows a color coding for the fuzzy membership of the computed clusters. Equation 5.62 can easily be generalized to a fuzzy membership function which is not only valid for the patterns but for the whole pattern space:

$$\mu_i(x) = \Big(\sum_{j=1}^{c} \big(\frac{||x - \pi_i||}{||x - \pi_j||}\big)^{\frac{2}{m-1}}\Big)^{-1} \tag{5.66}$$

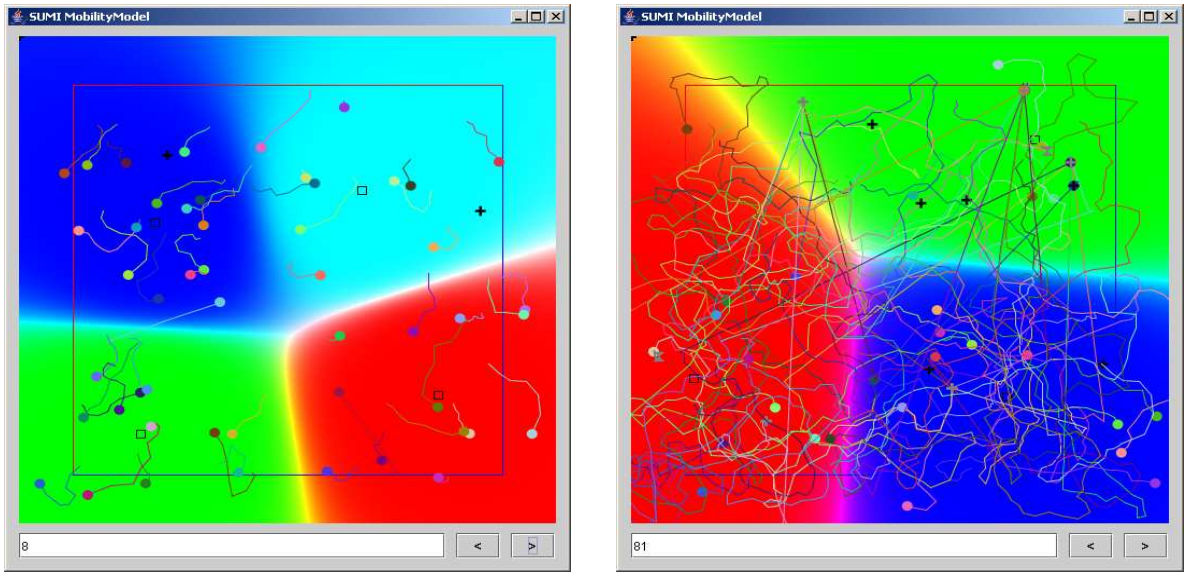


Figure 5.12: Simple Visualization of a SUMI Simulation Step with Fuzzy C-Means Ad-Hoc-Group detection. (50 nodes, $\alpha = 0.75$, area-dimensions: 500×500, number of days simulated: 3, iterations: 1024, shown iteration: no. 8 , number of active groups in this iteration: 0. Number of found groups: 4; shown other iteration: 81, number of active groups in this iteration: 4. Number of found groups: 3

We assign to every cluster a color (r_i, b_i, g_i), which is easily possible with the color sphere approach described in section 3.9. (The base colors of figure 5.12 were chosen according to a simpler scheme (see [37]).) We can then modulate this color in every point of the pattern space with the membership function $\mu_i(x)$. The overall color in point x is then computed as the sum of the color contributions from all clusters:

$$(r, g, b)(x) = \sum_{i=1}^{c} \mu_i(x)(r_i, b_i, g_i) \tag{5.67}$$

What figure 5.12 shows, is that due to the prototype based representation of the clusters together with the nearest neighbor like approach of computing the class memberships we get membership structures which are too delocalized to fulfill our expectation of a socially relevant cluster with respect to locations & velocities. Thus the crisp approach was chosen for this case as has been discussed in the previous sections.

5.2.1.3 Other Fuzzy Clustering Approaches

The alternating optimization approach can be conducted with several other expressions for the computation of U and Π. These expressions may either result from an optimization of a different objective function or may be direct evolvements from equations (5.62) and (5.63) for which no corresponding objective function may be found [161].
This is e.g. the case for the **Gustavson-Kessel-Clustering Algorithm**. In contrast to the Fuzzy C-Mean Algorithm, which finds spherical clusters, the Gustavson Kessel variant allows for finding ellipsoidal clusters. This is achieved by using a variant of the **Mahalanobis norm**

instead of the usual Euclidean norm. Using the Covariance matrix S of a cluster $\mathcal{C}_i$ [161]

$$S^{(i)} = \sum_{k=1}^{n} U_{ik}^m (x_k - \pi_i)(x_k - \pi_i)^{\mathrm{T}} \tag{5.68}$$

we can set for the norm $||\ ||$ in equation (5.62)

$$||x_k - \pi_i||^2 \to (\det(S_i))^{\frac{1}{\dim \mathcal{X}}} (x_k - \pi_i)^{\mathrm{T}} S^{(i)^{-1}} (x_k - \pi_i) \tag{5.69}$$

this corresponds to the Gustavson Kessel objective function [161]

$$J_{GK} = \sum_{k=1}^{n} \sum_{i=1}^{c} U_{ik}^m (\det(S_i))^{\frac{1}{\dim \mathcal{X}}} (x_k - \pi_i)^{\mathrm{T}} S^{(i)^{-1}} (x_k - \pi_i) \quad . \tag{5.70}$$

Another well known option is the possibilistic version of (5.62) [161] which leads to clusterings which do not satisfy condition (5.60).

$$U_{ik} = \left(1 + (\eta_i^{-1} ||x_k - \pi_i||)^{\frac{2}{m-1}}\right)^{-1} \tag{5.71}$$

This expression results from optimizing a slightly modified version of J_{GSQE} (equation (5.59)) defined by Krishnapuram and Keller (see [162]).

This membership function (generalizing x_k to x) is roughly smoothly cone shaped and the width is controlled (for each cluster) by the parameter η_i. Using this membership function avoids a consequence from condition (5.60) that appears less desirable in some cases: An outlier (a pattern $x_{k'}$ that lies way apart from the other patterns (that is for which $\forall i, k ||x_{k'} - \pi|| >> ||x_k - \pi||$ holds)) will approximately have membership degrees of $U_{ik'} \approx 1/c$ for all clusters due to condition (5.60) no matter how far it is apart from the other patterns. A pattern x_k that lies within the usual distance from the other patterns will usually have values $U_{ik} < 1/c$ for some clusters also because of condition (5.60) although the distance $||x_k - \pi_i||$ to the prototypes of these clusters may be much smaller than the distance $||x_{k'} - \pi_i||$ of the outlier to these clusters. This effect may or may not be desired depending on the application. The expression (5.71) avoids this effect.
In cases where no objective function can be formulated, we speak of **Alternating Cluster Estimation** (ACE) instead of Alternating Cluster Optimization. [162]

5.2.1.4 RACE

If we want to cluster our two interest test-sets consisting of 100 sets of free text interest phrases and 100 list-of-choice interest vectors, a problem arises that is of special significance for these types of data. Since we have no continuous metric space which the patterns lie in, we cannot apply the standard fuzzy clustering algorithms. What we have is a **relational matrix** R between the patterns which can either be a similarity matrix S or a distance matrix D. We assume (as before) that $S_{ik} = 1/(1 + D_{ik})$. As has been discussed at the beginning of 5.2, in our case we have a similarity matrix.
The idea to deal with this situation is to chose **patterns as cluster prototypes** and then using a variant of the AO-algorithm to iteratively optimize the matrix U_{ik} and the choices of the cluster prototypes on basis on the relational matrix only. The Starting from a predetermined number

RACE (Relational Alternating Cluster Estimation) Part I

- Input:
  - Number of clusters c.
  - Set of patterns $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$
  - Maximum number of iterations $t_{\max}$
  - A matrix of similarities $S \in [0,1]^{n \times n} : S_{k_1,k_2} = \mathtt{sim}(x_{k_1}, x_{k_2})$
- Initialize:
  - Determine randomly the initial set of prototype-patterns $\Pi^{(0)} = \{x_{k_1}^{(0)}, x_{k_2}^{(0)}, \ldots, x_{k_c}^{(0)}\}$
- Process:
  - For (step $t := 0$; $t \leq t_{\max}$; t++)
    - Compute $U^{(t)}(\Pi^{(t-1)}, S)$ using

$$U_{ik} = \Big(\sum_{j=1}^{c} \big(\frac{(1 - S_{k_i k}) S_{k_j k}}{(1 - S_{k_j k}) S_{k_i k}} \big)^{\frac{2}{m-1}} \Big)^{-1} \quad \text{(probabilistic case, similarity matrix } S\text{)} \tag{5.72}$$

OR

$$U_{ik} = \Big(\sum_{j=1}^{c} \big(\frac{D_{k_i k}}{D_{k_j k}} \big)^{\frac{2}{m-1}} \Big)^{-1} \quad \text{(probabilistic case, distance matrix } D\text{)} \tag{5.73}$$

OR

$$U_{ik} = \Big(1 + \big(\eta_i^{-1} (\frac{1}{S_{k_i k}} - 1) \big)^{\frac{2}{m-1}} \Big)^{-1} \quad \text{(possibilistic case, similarity matrix } S\text{)} \tag{5.74}$$

OR

$$U_{ik} = \Big(1 + \big(\eta_i^{-1} D_{k_i k} \big)^{\frac{2}{m-1}} \Big)^{-1} \quad \text{(possibilistic case, distance matrix } D\text{)} \tag{5.75}$$

Figure 5.13: RACE part I (adapted to our purposes from [161])

of clusters c and a randomly chosen set of prototype Patterns Π, each iteration determines the indices of those patterns that are used as cluster prototypes.
The algorithm is depicted in figure 5.14. The Algorithm uses the expression 5.62 for U_{ik}

RACE (Relational Alternating Cluster Estimation) Part II

* Compute $\Pi^{(t)}(U^{(t)})$ by the following steps:
  - randomly choose $x_{k_a} \in \Pi^{(t-1)}$
  - Compute the "energies" $E_{k_a,k_b} = \sum_{j \neq a} U_{jk_b}$.
    (In the probabilistic case we have $E_{k_a,k_b} = 1 - U_{ak_b}$.)
  - To get $\Pi^{(t)}$, replace $x_{k_a} \in \Pi^{(t-1)}$
    with the pattern that would yield the minimal energy if used as cluster prototype:
    $k_a \rightarrow \text{argmin}_{k_b \neq k_a}(E_{k_a,k_b})$.
    (In the probabilistic case we have $k_a \rightarrow \text{argmax}_{k_b \neq k_a}(U_{a,k_b})$.).

- Output:
  - Partition-Matrix $U \in [0,1]^{c \times n}$
    and cluster prototype-patterns $\Pi = \{x_{k_1}, x_{k_2}, \ldots, x_{k_c}\}$.

Figure 5.14: RACE part II (adapted to our purposes from [161])

substituting for the distances $||x - \pi||$ the relational expression

$$\text{sim}(a,b) = \frac{1}{1 + \text{dist}(a,b)} \quad \Rightarrow \quad ||x - \pi|| \rightarrow \frac{1}{\text{sim}(x,\pi)} - 1 \tag{5.76}$$

Alternatively we might use the possibilistic expression for U_{ik} given in equation (5.71).
The idea behind the energy-term E_{ik} is that the choice of a prototype k for cluster i implies an "energy" $E_{ik} = \sum_{j \neq i} U_{jk}$ so that minimizing the energy means that the degree of membership of a cluster's prototype in other clusters should be as small as possible. Every cluster should minimize inter-cluster coherence (and maximize intra-cluster coherence (which is not used here)).

5.2.1.5 Cluster Validation for Fuzzy Clusterings

In order to find the best number of clusters, **cluster validation** measures need to be employed. As has been discussed before, there are countless variants of such cluster validation measures. We will experiment with two of these approaches. The first well known cluster validation objective function is given by the **Partition Coefficient** [11, 60],

$$PC(c) = \frac{1}{n} \sum_{k=1}^{n} \sum_{i=1}^{c} U_{ik}^2 \tag{5.77}$$

The range of values is $PC(c) \in [1/c, 1]$ where the lower value is taken when complete fuzziness occurs ($\forall k, i \; U_{ik} = 1/c$) and the upper value is taken when we have a crisp clustering ($U_{ik} = 1$ for $k = k_i$ and $U_{ik} = 0$ for $k \neq k_i$). Therefore we can use the PC as a measure of fuzziness of the clustering or as a measure of how significantly each pattern is assigned to a specific cluster.

The more significantly the patterns are assigned to specific clusters (although the clustering is fuzzy and would in principle allow for a very "fuzzy" assignment) the better the clustering is assumed to be. It is clear that if the data itself has no detectable cluster structure, the index will also deliver small values.
The Partition Coefficient is therefore used to obtain the optimum cluster number via

$$c = \text{argmax}_{c'} PC(c') \quad . \tag{5.78}$$

The second validation criterion that we investigated is the **Partition Entropy** [60],

$$PE(c) = -\frac{1}{n} \sum_{k=1}^{n} \sum_{i=1}^{c} U_{ik} \ln U_{ik} \tag{5.79}$$

which measures the expectation value of the information of the distribution U_{ik} or equally (as it is common use in physics) the disorder of the distribution. The smaller the disorder of a distribution, the lower the entropy will be. For a crisp clustering (minimal disorder) we have a value of zero and for the maximally fuzzy clustering with $\forall k, i U_{ik} = 1/c$ we have a value of $\ln c$, thus $PE(c) \in [0, \ln c]$.
The clearer the cluster structure is that is reflected in U_{ik}, the better the clustering is assumed to be. Thus we have to obtain the optimal cluster number via

$$c = \text{argmin}_{c'} PE(c') \quad . \tag{5.80}$$

5.2.2 Applying Fuzzy Clustering

Before discussing how to apply the discussed techniques to the interest sets and vectors, a look will be taken at how the **general performance** of Fuzzy C-Means and RACE on artificially created data is, because especially for the RACE algorithm no experimental data and no comparative studies were available.
In order to see how the two algorithms performed compared to each other, a small application was written that delivers pattern matrices, similarity matrices and distance matrices of 2-dimensional geometrical data, which can be graphically input via mouse clicks. The tool can also be used to create artificial 4-dim. input data for the locations- and velocity investigations. The speed vectors are input via mouse drags. We used the tool to create the 3 matrices for a set of 20 patterns which where clearly structured in four distinct clusters. The patterns (points) are shown in figure 5.15. We ran the Fuzzy-C-Means algorithm 10 times with $m = 2$ on the 20×2 pattern matrix and the RACE algorithm in four variants each 10 times. The variants were

- (**variant s/po**) 20×20 similarity matrix and possibilistic calculation of memberships (equation (5.74)),
- (**variant s/pr**) 20×20 similarity matrix and probabilistic calculation of memberships (equation (5.72))
- (**variant d/po**) 20×20 distance matrix and possibilistic calculation of memberships (equation (5.75))
- (**variant d/pr**) 20×20 distance matrix and probabilistic calculation of memberships (equation (5.73)).

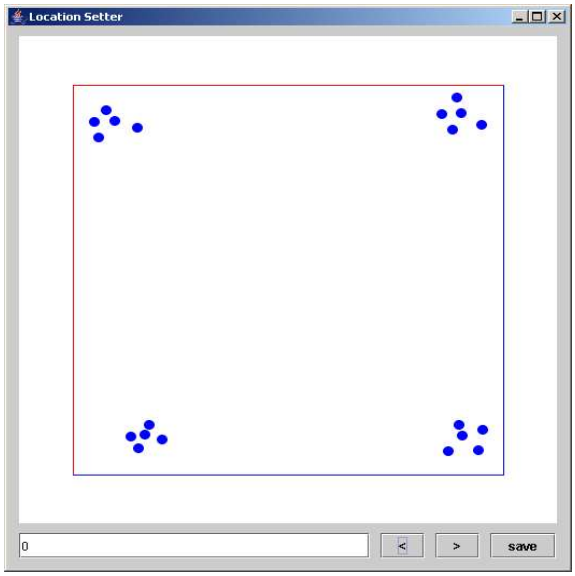


Figure 5.15: Creating absolute and relative input patterns for the Fuzzy Clustering algorithms with the location setter.

The results are shown in table 5.8.
Before discussing the results, a few words must be made on how the algorithms were implemented. We will then introduce how the experiment was conducted and thus how table 5.8 is to be read. Finally we will discuss the implications for our purposes.
The RACE algorithm was published by Runkler et al. in [163] with the presented update strategy for the cluster prototypes with the so called energies. The equations (5.74) and (5.72) use a similarity matrix converted into distances for the update of the membership matrix. Both variants of the **probabilistic update equations for** U_{ik} have a **problem** that has not been discussed in the publication [163] nor in the book [161]. In the case of the similarity matrix, the problem is that having $S_{k_i k} = 1$ (which is clearly the case for all prototype patterns themselves $(k_i = k)$) leads to $U_{k_i k} = \infty$. In our interpretation we set in these cases $U_{ik} = 1$. The same problem occurs in the distance matrix case for $D_{k_i k} = 0$. While this seems to be a minor problem at first glance, it turns out that it becomes a major problem in the course of updating the cluster prototypes. Since the heuristic leads to the update $k_a \rightarrow \text{argmin}_{k_b \neq k_a}(E_{k_a,k_b})$ which is in the probabilistic case $k_a \rightarrow \text{argmax}_{k_b \neq k_a}(U_{a,k_b})$, we always replace the current cluster prototype with itself, which is especially apparent in the probabilistic case, because $U_{a,k_b} = 1$ for $b = a$ but also occurs in the possibilistic variant. The attempt to circumvent this by taking not the argmin but the pattern with second smallest energy leads to a (irregular) oscillation between two patterns as prototypes for one cluster. Sometimes we find in our example scenario the case of (irregular) cyclic oscillation between all patterns of a cluster.
The great **problem of the proposed cluster prototype update strategy** that becomes apparent through this finding is that in the case of a clear cluster structure as in our testing scenario, the cluster prototypes can on principle **only change within** the cluster. If we thus have two cluster prototypes initially randomly assigned within one cluster, the update mechanism can never separate the two clusters assigned to these prototypes because they are actually only one cluster in reality. Thus we have an **extreme dependence on the initial random**

setting of the cluster prototypes in the proposed original algorithm. The experiments reflect

	# effectively found clusters	Fuzzy C-Means	RACE			
			s/po	s/pr	d/po	d/pr
correct	4	10	0	2	1	1
incorrect	3	0	8	6	7	8
	2	0	2	2	2	1
	1	0	0	0	0	0

Table 5.8: Running Fuzzy C-Means and RACE on the test data with $m = 2$ and $c = 4$.

# effectively found clusters	RACE overall	Relative Frequency of Occurrence	Theoretical Probability p_{draw}
4	4	0.10	0.1290
3	29	0.72	0.6192
2	7	0.18	0.2477
1	0	0	0.0041
$\sum$	40	1.0	1.0

Table 5.9: RACE and Drawing from samples

these problems very well.
Table 5.8 shows 10 runs of each of the four variants of RACE and 10 runs for Fuzzy-C-Means. The Fuzzy-C-Means could be tested on the experiment data of figure 5.15 because all three matrices (pattern-vectors, distance-matrix and similarity-matrix) were available for the experiment data. This is in contrast to the interest data for which no metric space exists and only relational data are available. In case of the **Fuzzy-C-Means** the resulting membership matrix reflected the real cluster structure of four almost crisply separated clusters in the test data **very well**. The membership values for points in each cluster were close to one, whereas the membership values for these points with respect to the other three clusters were very small (close to zero). Thus Fuzzy-C-Means was able to identify the correct cluster structure with **perfect "accuracy"**. In case of the RACE variants, no substantial differences could be noted between the four update variants of the membership update equations. **RACE always performed very bad**. We voted each run with respect to the number of real clusters found: If all four cluster prototypes were each placed in the four different real clusters, we counted this as four real clusters found. If two cluster prototypes where placed in the same cluster and the two others each in distinct other clusters, we voted this as three real clusters found and so on. Additionally, RACE always terminates when the maximum number of iterations is reached in contrast to Fuzzy-C-Means which has a termination criterion based on convergence. The nature of the prototype updating strategy of RACE leads to the same constant oscillating fluctuations in the cluster prototype assignment until $tmax$ is reached. Clearly this is not a convergent and desirable behavior.
To check our main **counterargument against the energy based prototype updating strategy of RACE** which is based on the theoretic considerations discussed above, we checked how the results of RACE corresponded with theory. The outcome is shown in table **Table**

5.9. If we randomly draw four numbers without replacement from a set of 20 numbers, where the numbers from 0 to 4 have color blue, numbers 5 to nine have color red, numbers 10 to 14 have color green and numbers 15 to 19 have color yellow, we get the probabilities shown in the last column of table 5.9. of drawing four numbers with either four, three, two or one different colors. These probabilities can be obtained after thorough combinatoric calculations which are not shown here and were checked through numeric simulation. This stochastic experiment corresponds to the initial random assignment of the cluster prototypes in RACE. The counter-argument that the energy based prototype updating strategy of RACE does not allow for an "escaping" of a "wrong" initial random assignment of the cluster prototypes with respect to the real clusters **is confirmed by the experiments**: The final cluster assignment well reflects the theoretical probability distribution of the initial random assignment of the cluster prototypes. That means that in case of such 4 distinct clusters, RACE cannot "escape" its initial assignment.

5.2.2.1 Finetuning RACE with Simulated Annealing

Viewing the problems of the original energy based prototype updating strategy of RACE from the point of view of search algorithms, we could say that the algorithm is not able to escape local minima in terms of searching for global minima of a cluster objective function like J_{GSQE}. In Artificial Intelligence many approaches have been developed to overcome such disadvantageous properties of a "greedy" optimization or search strategy. One well known strategy is **Simulated Annealing** (see e.g. [159]). In each iterative optimization or searching step, a new value is proposed and accepted with a certain probability. Greedy algorithms (like steepest ascent method) only accept (or propose) new values when they result in better values of the objective function. The idea of Simulated Annealing is roughly to propose new values randomly and to accept them with a certain probability if they lead to better values of the objective function ("ascending") but also, and this is the important trick, to accept them with a certain probability if they lead to a worse value of the objective function ("descending"). The probabilities of accepting the new values in either of the two cases are related and need to be dependent on time (or on the number of steps processed: The probability of accepting new values that lead to worse values of the objective function needs to be high at the beginning (in order to have a good chance of escaping local extremal values) and needs to decrease to zero at the end (in order to produce a convergent behavior and in order to reach the final extremal value). Usually, this probability is controlled by a parameter ϑ which is called **temperature**. Thus the whole process is called simulated annealing because of the analogy to a system from statistical physics which at high temperatures shows large fluctuations but reaches an energy minimum with small fluctuations when the temperature decreases.

We use the basic idea of the Simulated Annealing Technique to tune the energy based prototype updating strategy of RACE in a simple way. If a cluster has been randomly picked the energies E_{ik} are computed in the usual way. But instead of always picking the pattern with the minimal energy or the second minimal energy as new cluster prototype, we introduce a "temperature dependent" probability of choosing also patterns with larger energies. We define the temperature dependent set of probabilities

$$p_m(\vartheta) = \vartheta^m(1-\vartheta) + \frac{\vartheta^n}{n} \tag{5.81}$$

for choosing the pattern x_k with the $m-$th minimal energy E_{ik} as the new prototype for cluster $\mathcal{C}_i$. The 0-th minimal energy is the overall minimal energy (corresponding to $k = k_i$).

We have

$$\sum_{m=0}^{n-1} p_m(\vartheta) = n\frac{\vartheta^n}{n} + (1-\vartheta)\sum_{m=0}^{n-1}\vartheta^m = \vartheta^n + (1-\vartheta)\frac{1-\vartheta^n}{1-\vartheta} = 1 \tag{5.82}$$

and $p_m(0)$ yields the original prototype updating strategy of RACE.
The temperature should be chosen in $\vartheta \in [0,1]$.
If we have a random number $z \in [0,1]$ we can easily determine which pattern is used as the new prototype for the cluster. If we take the pattern with the y-th minimal energy we have to determine y by solving the equation

$$z \leq \sum_{m=0}^{y} p_m(\vartheta) \tag{5.83}$$

$$z \leq \vartheta^n\frac{y}{n} + 1 - \vartheta^{y+1} \tag{5.84}$$

Since $\vartheta^n\frac{y}{n} \ll 1$ we set

$$y = \left\lceil \frac{\ln(1-z)}{\ln\vartheta} - 1 \right\rceil \tag{5.85}$$

As figure 5.16 shows, we still keep the largest probability for choosing the pattern with absolute

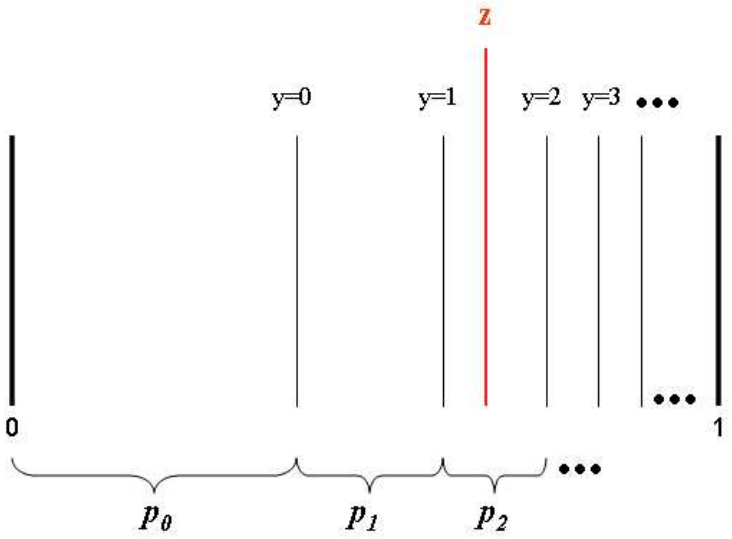


Figure 5.16: Determining the new prototype index from a random number.

(0-th) minimal energy and while the temperature decreases through the course of the simulation as

$$\vartheta(t) = \vartheta_0\frac{t}{t_{\max}} \tag{5.86}$$

the greedy standard behavior is approached. Convergence is detected by detecting for 30 iterations the same set of cluster prototypes. In this way, the algorithm is sped up by a factor of 10 in our experiments.
Using this adapted Simulated Annealing approach for the updating of the cluster prototypes results in an algorithm that will be called RACE-SA. The Performance of RACE-SA is substantially improved compared with the standard RACE as **table 5.10** shows.
Although RACE can now be applied to the problem of identifying groups in the relational matrices characterizing the relations between the interests of the users, the performance is still not perfect. We will now discuss a surprisingly simple approach to relational clustering which works very well for the test-data with the four clusters.

	# effectively found clusters	RACE-SA				Relative Frequency of Occurrence
		s/po	s/pr	d/po	d/pr	
correct	4	8	8	6	9	0.775
incorrect	3	2	2	4	1	0.225
	2	0	0	0	0	
	1	0	0	0	0	

Table 5.10: Performance of RACE-SA on the test data with $\vartheta_0 = 0.8$, $m = 2$ and $c = 4$.

5.2.2.2 RFAO-Relational Fuzzy Alternating Optimization

The RACE-SA approach works very well and has a substantial time performance benefit through the introduction of a convergent behavior of the algorithm. Nevertheless **another approach** to fuzzy clustering relational data has been tried which works extremely well and which we will now discuss.
The main aspect why RACE was developed was that for data where only relations (distance or similarity) between the patterns are known we do not have a metric space where the prototypes could be calculated independently from the patterns with an equation of type (5.63). However, what characterizes most metric spaces for patterns is that usually we have a set of m properties which characterize the patterns and which are associated with an numeric interval value. Thus the pattern space is very often a subset of $\mathbb{R}^m$. If we view the n **relations** of a pattern x_k to the other patterns x_l (a similarity S_{kl} or a distance D_{kl}) **as properties** we can view the relational space as a metric space (subspace of $\mathbb{R}^n$) and use standard fuzzy clustering algorithms on that space in the usual way. The n-dimensional rows (or columns) of the relational matrix become the patterns which characterize the individual objects to be clustered.
The clusters will then be characterized by a prototype which is another n-dimensional vector and whose components can be interpreted as relations to the n patterns.
Another approach is to use the excellent convergence and reliability properties of the "conventional" fuzzy clustering algorithm to compute memberships and prototypes on the relational "patterns" and then compute the patterns with the smallest distance to the prototypes. These patterns are taken as cluster prototypes (in the usual fashion of RACE) and the final membership matrix is then computed by a single application of one of the equations (5.72), (5.73), (5.74) or (5.75) depending on the nature of the relational data.

An important point to regard is that in order to use this idea on similarity data, these data have to be converted into distances via dist = 1/sim - 1 first. Using a Fuzzy-C-Means directly with the similarity as pattern matrix results in complete fuzziness (which is $\forall k, i\ U_{ik} = 1/c$) which was tested with 20 runs.
Using the Fuzzy-C-Means on our 4 cluster distance matrix as pattern matrix resulted in **100 % accuracy in 50 test runs**. This makes this new approach which we labled as **RFAO-Relational Fuzzy Alternating Optimization** another excellent candidate for identifying groups in the relational interest data.

5.2.2.3 Cluster Validation Results

In order to apply one of the two techniques (RACE-SA and RFAO) to the problem of finding groups, we need to shortly investigate, whether cluster validation techniques are able to determine the right number of clusters. This investigation can not be systematic because the number of available techniques is very large. We will therefore only test the two cluster validation strategies that were introduced in 5.2.1.5.
Table 5.11 shows the results. A cluster validation run involves running the clustering algorithm

	RFAO		RACE-SA s/po		RACE-SA s/pr	
	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect
PE	20	0	16	4	7	13
PC	20	0	15	5	14	6

Table 5.11: Performance of PC and PE of 20 runs with RACE-SA and RFAO on the four cluster test data with parameters $\vartheta_0 = 0.8$ and $m = 2$.

with numbers of clusters ranging from 2 to $n - 1$ and then determining which run was best with respect to the respective validation criterion. If the outcome of such a run was that the optimum number of clusters was four (on our four cluster test data) the validation was judged to be correct, otherwise incorrect.
We tested only the two variants of RACE-SA that work with similarity matrices since this will also be the case for the interest investigations. What we can see from table 5.11 is that with RFAO either of the two cluster validity measures delivers perfect results. In case of RACE-SA, we find that although 3 of the four combinations go well with each other, the Partition Entropy does not deliver good results when combined with a probabilistic membership update strategy in the clustering algorithm. This might be attributed to the fact that the probabilistic membership update function is less focused when it comes to differentiating the membership degrees of patterns that lie far away from the cluster center as has been explained in 5.2.1.3.
The consequence is that the cluster validation needs to be **run multiple times** and the optimal number of clusters is taken to be the cluster number with the most votes.

5.2.3 Results for Interest Data

We will now apply the considerations of the previous sections to the case of the interest sets and vectors. For the sake of simplicity we will use RACE-SA in s/po configuration with $m = 2$ and

$\vartheta_0 = 0.8$ only and will use PC as cluster validation tool. We also limited the test size from the 100×100 matrices to a subset of 20×20 matrices.

5.2.3.1 Free Text Interest Phrases

The test runs where conducted on the 20×20 matrices which resulted from applying the similarity measure to compare sets of free text interest phrases is shown in figures 4.9 and 4.10 to the first 20 sets of the test collection "Survey Collection" shown in figures C.1 and C.2 in the appendix. The similarities between these 20 elements were computed as:

"Survey Collection" (First 20 elements)

(1) reading; windsurfing; computers; lying at the sea; biking
(2) footbal; newspaper; music; mobile phones; sports; tv; movies; computers; women; drinking beer
(3) skiing; dancing; sailing; cycling; cinema; foreign languages; reading; programming
(4) volleyball; soccer; jogging; inline-skating; playing drums; drawing; meeting friends; visiting exibitions; reading books; spead reading; party; playing cards and board games; icq'ing; playing tennis
(5) sailing; books; tabletennis; juggling; friends; games
(6) skiing; listening to bach and beethoven; esp piano music; travel; food
(7) music; reading; books; cars; motorracing; swimming; hiking; cycling; hanging around; food; architecture; traveling; sleeping
(8) jogging; music; sports; mountains; traveling; computer; books; movies; sub-culture; theater; writing texts; friends; parties; festivals; art; painting
(9) cooking; kung fu; jogging; reading; watching tv; hiking; biking; swimming; singing; trash metal; cars; meeting friends; modern arts
(10) money; sex; wisdom; traveling; sea; sports; reading; computer; talking; drinking; movies; humor; childs
(11) computer; photo; model railway; football; music
(12) reading; books in general; traveling; especially oriental countries; religion; christianity; getting to know human beings; talking with friends and my family; cooking; good wine; roleplaying like DSA and D´n´D; YMCA; honarary works; camping; rollerblading; photography; dancing; singing; playing guitar; moving; sleeping; day-dreaming
(13) soccer; canyoning; beach volleyball; cycling; programming; java; frag georg; j2ee; michel friedman
(14) jazz improvisation; modern music; baroque music; lied; operating systems; linux; digital audio processing; information systems engineering; systems administration; networking; network security; free software; internet communities; contemporary english literature; poetry; theatre; modern art
(15) reading; chess; hiking; music; poetry; meditation
(16) soccer; badminton; boxing; music; good food; travelling; clubbing; going out for a drink; dancing; hanging loose; discussing; dreaming; hiking; going to the movies; thinking; reading
(17) baseball; sport; skiing; music; travelling; soccer; photography; tolkien
(18) reading books; going to the movies; playing golf; listening to music; being with monika; ballroom dancing
(19) music; zeitgeist; love; friends
(20) meeting friends; cinema; jogging; aerobic; listen music; dancing; reading; watching tv; star trek; visiting my family; musicals; cats

Figure 5.17: The first 20 free text interest phrases sets from the "Survey Collection"

$$S = \begin{pmatrix}
1.000 & 0.228 & 0.331 & 0.202 & 0.089 & 0.094 & 0.289 & 0.204 & 0.353 & 0.411 & 0.282 & 0.200 & 0.133 & 0.106 & 0.248 & 0.206 & 0.083 & 0.199 & 0.040 & 0.227 \\
0.228 & 1.000 & 0.258 & 0.163 & 0.147 & 0.263 & 0.252 & 0.433 & 0.196 & 0.451 & 0.411 & 0.153 & 0.115 & 0.248 & 0.220 & 0.283 & 0.296 & 0.329 & 0.262 & 0.304 \\
0.331 & 0.258 & 1.000 & 0.284 & 0.243 & 0.289 & 0.357 & 0.254 & 0.296 & 0.310 & 0.219 & 0.296 & 0.338 & 0.147 & 0.260 & 0.358 & 0.316 & 0.385 & 0.076 & 0.425 \\
0.202 & 0.163 & 0.284 & 1.000 & 0.320 & 0.197 & 0.246 & 0.362 & 0.338 & 0.220 & 0.258 & 0.271 & 0.257 & 0.127 & 0.217 & 0.295 & 0.287 & 0.358 & 0.176 & 0.386 \\
0.089 & 0.147 & 0.243 & 0.320 & 1.000 & 0.111 & 0.211 & 0.327 & 0.175 & 0.118 & 0.121 & 0.214 & 0.105 & 0.115 & 0.060 & 0.096 & 0.109 & 0.213 & 0.228 & 0.189 \\
0.094 & 0.263 & 0.289 & 0.197 & 0.111 & 1.000 & 0.491 & 0.352 & 0.180 & 0.290 & 0.238 & 0.259 & 0.139 & 0.180 & 0.200 & 0.394 & 0.515 & 0.238 & 0.223 & 0.222 \\
0.289 & 0.252 & 0.357 & 0.246 & 0.211 & 0.491 & 1.000 & 0.370 & 0.427 & 0.313 & 0.294 & 0.366 & 0.183 & 0.219 & 0.445 & 0.487 & 0.345 & 0.314 & 0.211 & 0.272 \\
0.204 & 0.433 & 0.254 & 0.362 & 0.327 & 0.352 & 0.370 & 1.000 & 0.269 & 0.383 & 0.394 & 0.274 & 0.136 & 0.332 & 0.206 & 0.301 & 0.336 & 0.396 & 0.356 & 0.378 \\
0.353 & 0.196 & 0.296 & 0.338 & 0.175 & 0.180 & 0.427 & 0.269 & 1.000 & 0.207 & 0.151 & 0.340 & 0.137 & 0.150 & 0.306 & 0.272 & 0.159 & 0.198 & 0.158 & 0.414 \\
0.411 & 0.451 & 0.310 & 0.220 & 0.118 & 0.290 & 0.313 & 0.383 & 0.207 & 1.000 & 0.287 & 0.292 & 0.099 & 0.176 & 0.259 & 0.357 & 0.279 & 0.318 & 0.137 & 0.267 \\
0.282 & 0.411 & 0.219 & 0.258 & 0.121 & 0.238 & 0.294 & 0.394 & 0.151 & 0.287 & 1.000 & 0.185 & 0.191 & 0.280 & 0.294 & 0.336 & 0.443 & 0.305 & 0.266 & 0.272 \\
0.200 & 0.153 & 0.296 & 0.271 & 0.214 & 0.259 & 0.366 & 0.274 & 0.340 & 0.292 & 0.185 & 1.000 & 0.096 & 0.115 & 0.243 & 0.337 & 0.276 & 0.300 & 0.154 & 0.288 \\
0.133 & 0.115 & 0.338 & 0.257 & 0.105 & 0.139 & 0.183 & 0.136 & 0.137 & 0.099 & 0.191 & 0.096 & 1.000 & 0.072 & 0.054 & 0.200 & 0.291 & 0.122 & 0.053 & 0.095 \\
0.106 & 0.248 & 0.147 & 0.127 & 0.115 & 0.180 & 0.219 & 0.332 & 0.150 & 0.176 & 0.280 & 0.115 & 0.072 & 1.000 & 0.299 & 0.175 & 0.205 & 0.195 & 0.264 & 0.190 \\
0.248 & 0.220 & 0.260 & 0.217 & 0.060 & 0.200 & 0.445 & 0.206 & 0.306 & 0.259 & 0.294 & 0.243 & 0.054 & 0.299 & 1.000 & 0.462 & 0.224 & 0.340 & 0.262 & 0.311 \\
0.206 & 0.283 & 0.358 & 0.295 & 0.096 & 0.394 & 0.487 & 0.301 & 0.272 & 0.357 & 0.336 & 0.337 & 0.200 & 0.175 & 0.462 & 1.000 & 0.420 & 0.449 & 0.197 & 0.327 \\
0.083 & 0.296 & 0.316 & 0.287 & 0.109 & 0.515 & 0.345 & 0.336 & 0.159 & 0.279 & 0.443 & 0.276 & 0.291 & 0.205 & 0.224 & 0.420 & 1.000 & 0.287 & 0.226 & 0.239 \\
0.199 & 0.329 & 0.385 & 0.358 & 0.213 & 0.238 & 0.314 & 0.396 & 0.198 & 0.318 & 0.305 & 0.300 & 0.122 & 0.195 & 0.340 & 0.449 & 0.287 & 1.000 & 0.211 & 0.397 \\
0.040 & 0.262 & 0.076 & 0.176 & 0.228 & 0.223 & 0.211 & 0.356 & 0.158 & 0.137 & 0.266 & 0.154 & 0.053 & 0.264 & 0.262 & 0.197 & 0.226 & 0.211 & 1.000 & 0.287 \\
0.227 & 0.304 & 0.425 & 0.386 & 0.189 & 0.222 & 0.272 & 0.378 & 0.414 & 0.267 & 0.272 & 0.288 & 0.095 & 0.190 & 0.311 & 0.327 & 0.239 & 0.397 & 0.287 & 1.000
\end{pmatrix} \tag{5.87}$$

The free text interest sets were investigated with the RACE-SA algorithm with possibilistic membership update strategy and Partition Coefficient as cluster-validation technique (parameters were $\vartheta = 0.8$ and $m = 2$) and with the RFAO algorithm with Partition Coefficient as cluster-validation technique ($m = 2$). Both strategies were run 50 times (each with c ranging from 2 to 19) and the number of classes with the most votes was taken as the overall best number of classes. Both strategies predicted an **overall best number of classes of 2**. (each with 50 votes). The algorithms were therefore finally run with $c = 2$ to obtain the membership matrix. In case of RACE SA the set of two prototype indices of 50 runs with the most votes was accepted as the prototype patterns and the membership matrix was computed in a single step.

The Results are shown in table 5.12. As can be seen from looking at table 5.12 and figure 5.17, the interests in this 20×20 part of Survey Collection **do not show** an extremely **distinct cluster structure**. What we can subjectively extract by looking at the interest sets is that we can divide them into such interest sets which are dominated by sports and action-rich, bodily activities and those interest sets which are dominated by contemplative occupations like reading or listening to music. This distinction is not reflected in the results of the clustering. RACE-SA prototypes 5 and 16 could only be interpreted as prototypes reflecting rather individual occupations (16) and rather socially oriented occupations (5) which is somehow also reflected in the membership degrees, presumably symbolized by the common element "friends".

What would have to be done in order to be able to quantitatively test the predictive power of the group detection strategies that have been introduced, is to produce a statistically relevant subjective clustering of the given test collection by a large number of users and then to compare the predictions of the group detection with the average cluster tagging by the survey participants. Since this would require substantial efforts, it would by far exceed the scope of this thesis

Membership-Matrix U_{ik} RACE-SA	$\begin{pmatrix} 0.13 & 0.16 & 0.26 & 0.56 & 1.00 & 0.04 & 0.09 & 0.56 & 0.25 & 0.06 & 0.07 & 0.23 & 0.18 & 0.28 & 0.01 & 0.00 & 0.03 & 0.11 & 0.59 & 0.19 \\ 0.87 & 0.84 & 0.74 & 0.44 & 0.00 & 0.96 & 0.91 & 0.44 & 0.75 & 0.94 & 0.93 & 0.77 & 0.82 & 0.72 & 0.99 & 1.00 & 0.97 & 0.89 & 0.41 & 0.81 \end{pmatrix}$
Prototype Pattern Indicies k_1 and k_2 RACE-SA	4 and 15
Membership-Matrix U_{ik} RFAO	$\begin{pmatrix} 0.55 & 0.25 & 0.40 & 0.34 & 0.62 & 0.40 & 0.19 & 0.23 & 0.35 & 0.24 & 0.25 & 0.35 & 0.60 & 0.56 & 0.46 & 0.22 & 0.41 & 0.14 & 0.55 & 0.21 \\ 0.45 & 0.75 & 0.60 & 0.66 & 0.38 & 0.60 & 0.81 & 0.77 & 0.65 & 0.76 & 0.75 & 0.65 & 0.40 & 0.44 & 0.54 & 0.78 & 0.59 & 0.86 & 0.45 & 0.79 \end{pmatrix}$
Prototype Vectors π_1 and π_2 RFAO	$\begin{pmatrix} 7.37 & 3.94 & 3.62 & 3.41 & 6.29 & 4.63 & 2.76 & 2.81 & 3.95 & 4.36 & 3.46 & 4.43 & 8.07 & 5.63 & 5.98 & 3.69 & 4.15 & 3.51 & 7.25 & 3.66 \\ 4.43 & 2.70 & 2.51 & 2.77 & 5.67 & 2.96 & 2.02 & 1.97 & 3.14 & 2.54 & 2.56 & 2.99 & 6.81 & 4.47 & 3.06 & 2.05 & 2.65 & 2.17 & 4.75 & 2.30 \end{pmatrix}$

Table 5.12: Results of Group Detection on 20×20 Part of the Survey Collection.

5.2.3.2 List-Of-Choice Interest Vectors

The test collection "Dating Collection" of list-of-choice interest vectors is shown in figures C.3 and C.4 in the appendix. The similarity measure to compare sets of free text interest phrases is shown in figures 4.9 and 4.10 and the similarity measure to compare list-of-choice interest vectors is shown in figure 4.5. The similarities between these 20 elements were computed as:

"Survey Collection" (First 20 elements)

(1) Dancing, Family, Movies, Listening to Music, Reading, Watching Sports, Theater, Travel, Cooking, Health/Fitness
(2) Arts, Dancing, Dining, Family, Movies, Outdoor Activities, Photography, Watching Sports, Theater, Travel, Cooking, Computers / Internet, Television, Gardening, Playing Music, Playing Sports, Health/Fitness
(3) Dining, Family, Movies, Listening to Music, Reading, Theater, Travel
(4) Arts, Movies, Listening to Music, Reading, Theater, Travel, Cooking, Computers / Internet, Health/Fitness
(5) Arts, Dancing, Dining, Movies, Listening to Music, Outdoor Activities, Photography, Reading, Theater, Travel, Cooking, Computers / Internet, Television, Crafts, Health/Fitness
(6) Arts, Dancing, Listening to Music, Outdoor Activities, Photography, Reading, Travel
(7) Dancing, Dining, Family, Listening to Music, Outdoor Activities, Photography, Reading, Religion / Spirituality, Theater, Travel, Cooking, Crafts, Playing Music, Health/Fitness
(8) Family, Movies, Listening to Music, Reading, Computers / Internet
(9) Arts, Community Service, Dining, Family, Movies, Listening to Music, Photography, Reading, Theater, Travel
(10) Arts, Dining, Family, Movies, Outdoor Activities, Reading, Travel
(11) Arts, Community Service, Dancing, Dining, Family, Listening to Music, Reading, Religion / Spirituality, Theater, Travel, Cooking, Gardening, Health/Fitness
(12) Arts, Dining, Movies, Listening to Music, Outdoor Activities, Reading, Travel, Cooking
(13) Arts, Family, Movies, Listening to Music, Outdoor Activities, Travel, Cooking, Computers / Internet
(14) Dancing, Dining, Movies, Listening to Music, Reading, Watching Sports, Travel, Television, Health/Fitness
(15) Dancing, Dining, Outdoor Activities, Crafts
(16) Dancing, Dining, Movies, Listening to Music, Outdoor Activities, Theater, Travel, Television
(17) Arts, Family, Movies, Listening to Music, Outdoor Activities, Photography, Reading, Theater, Travel
(18) Dancing, Dining, Movies, Reading, Cooking
(19) Movies, Listening to Music, Reading, Theater, Travel, Computers / Internet, Television
(20) Dancing, Movies, Listening to Music, Travel, Playing Music

Figure 5.18: The first 20 List-Of-Choice interest vectors from the "Dating Collection"

$$S = \begin{pmatrix}
1.000 & 0.771 & 0.762 & 0.792 & 0.777 & 0.644 & 0.796 & 0.556 & 0.672 & 0.669 & 0.789 & 0.703 & 0.718 & 0.851 & 0.433 & 0.733 & 0.724 & 0.724 & 0.633 & 0.693 \\
0.771 & 1.000 & 0.575 & 0.687 & 0.848 & 0.639 & 0.782 & 0.410 & 0.607 & 0.672 & 0.711 & 0.649 & 0.736 & 0.743 & 0.578 & 0.737 & 0.666 & 0.638 & 0.542 & 0.589 \\
0.762 & 0.575 & 1.000 & 0.756 & 0.686 & 0.530 & 0.681 & 0.711 & 0.887 & 0.789 & 0.718 & 0.763 & 0.699 & 0.698 & 0.254 & 0.750 & 0.822 & 0.653 & 0.789 & 0.646 \\
0.792 & 0.687 & 0.756 & 1.000 & 0.852 & 0.670 & 0.656 & 0.666 & 0.750 & 0.683 & 0.701 & 0.809 & 0.806 & 0.702 & 0.214 & 0.654 & 0.798 & 0.651 & 0.821 & 0.618 \\
0.777 & 0.848 & 0.686 & 0.852 & 1.000 & 0.775 & 0.819 & 0.531 & 0.712 & 0.713 & 0.718 & 0.804 & 0.771 & 0.804 & 0.603 & 0.812 & 0.773 & 0.734 & 0.730 & 0.617 \\
0.644 & 0.639 & 0.530 & 0.670 & 0.775 & 1.000 & 0.715 & 0.462 & 0.643 & 0.688 & 0.635 & 0.752 & 0.683 & 0.665 & 0.503 & 0.612 & 0.785 & 0.554 & 0.508 & 0.613 \\
0.796 & 0.782 & 0.681 & 0.656 & 0.819 & 0.715 & 1.000 & 0.452 & 0.670 & 0.667 & 0.835 & 0.695 & 0.691 & 0.699 & 0.651 & 0.683 & 0.694 & 0.650 & 0.479 & 0.630 \\
0.556 & 0.410 & 0.711 & 0.666 & 0.531 & 0.462 & 0.452 & 1.000 & 0.713 & 0.553 & 0.469 & 0.500 & 0.674 & 0.461 & 0.000 & 0.409 & 0.734 & 0.404 & 0.745 & 0.499 \\
0.672 & 0.607 & 0.887 & 0.750 & 0.712 & 0.643 & 0.670 & 0.713 & 1.000 & 0.791 & 0.763 & 0.732 & 0.721 & 0.600 & 0.204 & 0.631 & 0.891 & 0.569 & 0.713 & 0.550 \\
0.669 & 0.672 & 0.789 & 0.683 & 0.713 & 0.688 & 0.667 & 0.553 & 0.791 & 1.000 & 0.712 & 0.863 & 0.780 & 0.665 & 0.471 & 0.650 & 0.807 & 0.704 & 0.544 & 0.493 \\
0.789 & 0.711 & 0.718 & 0.701 & 0.718 & 0.635 & 0.835 & 0.469 & 0.763 & 0.712 & 1.000 & 0.705 & 0.685 & 0.673 & 0.463 & 0.606 & 0.663 & 0.655 & 0.475 & 0.519 \\
0.703 & 0.649 & 0.763 & 0.809 & 0.804 & 0.752 & 0.695 & 0.500 & 0.732 & 0.863 & 0.705 & 1.000 & 0.822 & 0.727 & 0.466 & 0.722 & 0.758 & 0.785 & 0.615 & 0.625 \\
0.718 & 0.736 & 0.699 & 0.806 & 0.771 & 0.683 & 0.691 & 0.674 & 0.721 & 0.780 & 0.685 & 0.822 & 1.000 & 0.590 & 0.366 & 0.655 & 0.796 & 0.578 & 0.636 & 0.630 \\
0.851 & 0.743 & 0.698 & 0.702 & 0.804 & 0.665 & 0.699 & 0.461 & 0.600 & 0.665 & 0.673 & 0.727 & 0.590 & 1.000 & 0.534 & 0.821 & 0.609 & 0.748 & 0.656 & 0.717 \\
0.433 & 0.578 & 0.254 & 0.214 & 0.603 & 0.503 & 0.651 & 0.000 & 0.204 & 0.471 & 0.463 & 0.466 & 0.366 & 0.534 & 1.000 & 0.578 & 0.251 & 0.570 & 0.069 & 0.354 \\
0.733 & 0.737 & 0.750 & 0.654 & 0.812 & 0.612 & 0.683 & 0.409 & 0.631 & 0.650 & 0.606 & 0.722 & 0.655 & 0.821 & 0.578 & 1.000 & 0.684 & 0.663 & 0.729 & 0.747 \\
0.724 & 0.666 & 0.822 & 0.798 & 0.773 & 0.785 & 0.694 & 0.734 & 0.891 & 0.807 & 0.663 & 0.758 & 0.796 & 0.609 & 0.251 & 0.684 & 1.000 & 0.514 & 0.755 & 0.596 \\
0.724 & 0.638 & 0.653 & 0.651 & 0.734 & 0.554 & 0.650 & 0.404 & 0.569 & 0.704 & 0.655 & 0.785 & 0.578 & 0.748 & 0.570 & 0.663 & 0.514 & 1.000 & 0.487 & 0.550 \\
0.633 & 0.542 & 0.789 & 0.821 & 0.730 & 0.508 & 0.479 & 0.745 & 0.713 & 0.544 & 0.475 & 0.615 & 0.636 & 0.656 & 0.069 & 0.729 & 0.755 & 0.487 & 1.000 & 0.620 \\
0.693 & 0.589 & 0.646 & 0.618 & 0.617 & 0.613 & 0.630 & 0.499 & 0.550 & 0.493 & 0.519 & 0.625 & 0.630 & 0.717 & 0.354 & 0.747 & 0.596 & 0.550 & 0.620 & 1.000
\end{pmatrix} \tag{5.88}$$

Clustering these free text interest phrases with the same methods yields the results shown in table 5.13. Again, as in the case of the free test interest sets, the Collection does subjectively not show a very prominent cluster structure. In both cases, the PC cluster validation predicts an overall best number of clusters of 2 which may be a coincidence but which could also be a general tendency of the PC based method if no distinct clusters are present.

Membership-Matrix U_{ik} RACE-SA	$\begin{pmatrix} 0.13 & 0.16 & 0.26 & 0.56 & 1.00 & 0.04 & 0.09 & 0.56 & 0.25 & 0.06 & 0.07 & 0.23 & 0.18 & 0.28 & 0.01 & 0.00 & 0.03 & 0.11 & 0.59 & 0.19 \\ 0.87 & 0.84 & 0.74 & 0.44 & 0.00 & 0.96 & 0.91 & 0.44 & 0.75 & 0.94 & 0.93 & 0.77 & 0.82 & 0.72 & 0.99 & 1.00 & 0.97 & 0.89 & 0.41 & 0.81 \end{pmatrix}$
Prototype Pattern Indicies k_1 and k_2 RACE-SA	4 and 15
Membership-Matrix U_{ik} RFAO	$\begin{pmatrix} 0.55 & 0.25 & 0.40 & 0.34 & 0.62 & 0.40 & 0.19 & 0.23 & 0.35 & 0.24 & 0.25 & 0.35 & 0.60 & 0.56 & 0.46 & 0.22 & 0.41 & 0.14 & 0.55 & 0.21 \\ 0.45 & 0.75 & 0.60 & 0.66 & 0.38 & 0.60 & 0.81 & 0.77 & 0.65 & 0.76 & 0.75 & 0.65 & 0.40 & 0.44 & 0.54 & 0.78 & 0.59 & 0.86 & 0.45 & 0.79 \end{pmatrix}$
Prototype Vectors π_1 and π_2 RFAO	$\begin{pmatrix} 7.37 & 3.94 & 3.62 & 3.41 & 6.29 & 4.63 & 2.76 & 2.81 & 3.95 & 4.36 & 3.46 & 4.43 & 8.07 & 5.63 & 5.98 & 3.69 & 4.15 & 3.51 & 7.25 & 3.66 \\ 4.43 & 2.70 & 2.51 & 2.77 & 5.67 & 2.96 & 2.02 & 1.97 & 3.14 & 2.54 & 2.56 & 2.99 & 6.81 & 4.47 & 3.06 & 2.05 & 2.65 & 2.17 & 4.75 & 2.30 \end{pmatrix}$

Table 5.13: Results of Group Detection on 20×20 Part of the Dating Collection.

5.3 Detection and Modeling of Groups on the Basis of Communication Patterns

As an example for identifying groups with respect to communication behavior, we have discussed in section 4.8 how to compare two users that **communicate via a medium with a tree-like structure**. Such a medium is realized in form of the well known Newsgroups and various related forms of discussion boards on the web. We have proposed a similarity measure which was motivated by the **principles** that the social relatedness of two persons is assumed to be greater the

- **more** discrete communication acts (postings, board contributions) are addressed by a person as a reaction to a discrete communication act by another person.
- the more **mutual** communication acts there are.

These principles have been discussed and motivated in section 4.8 and led to the construction of a similarity measure (4.30). This similarity measure again produces (as in the case of the interests) a similarity matrix.
In terms of **clustering** on the basis of the communication behavior and content of mobile community members the same arguments as in the case of their interests apply: A crisp clustering would produce artificial results, because the nature of communication structure and especially communication content is more suited to be described with the help of **fuzzy** sets. Again, since we do not have a "continuous" metric space we must also use clustering algorithms which are able to solely work on relational data such as similarities.
We will now shortly discuss the test data and qualitative performance of our similarity measure (4.30) in connection with the relational clustering algorithms that werw introduced in the previous sections.

5.3.1 Test-Data and Coarse Qualitative Performance of Relational Clustering Algorithms

2000 postings from four newsgroups were downloaded (September 2004), preprocessed and the similarity matrices between the users of each of the four newsgroups were computed with the similarity measure (4.30). Table 5.14 gives an overview of the newsgroups that were used for the test-runs. The preprocessing was done with the Rainbow toolkit [114] and for the communication with the News-Server, a freeware NNTP client [140] was used. What we can see is that the news postings are not always replies to other postings which is often the case with net news. In community discussion boards we find a more tree like structure but these boards are not as easily accessible to crawlers therefore an NNTP based crawler and analyzer was created. The similarity matrix computation extracts the reply matrix and then implements the similarity measure (4.30).
Applying the RACE-SA and RFAO algorithm to these similarity matrices produces clusters which can only be analyzed through a very laborious subjective process. Without being able to systematically discuss this analysis process, one can look at the posting contents and reply relations of the main cluster members and compare them qualitatively among each other. The result is that the relational cluster methods perform reasonably well with the similarity measure (4.30) on the four newsgroups. What we can see is that the cluster structures are more

Name	Subject	No. of Users	No. of Postings	Dim. Vocabulary	Total No. of Replies
alt.underground @freenews.netfront.net	Underground Music and Lifestyle	28	500	4518	411
uk.people.gothic @freenews.netfront.net	Gothic Music and Lifestyle	111	500	6970	293
uk.rec.audio @freenews.netfront.net	Stereo Hardware	70	500	5271	395
uk.rec.cycling @freenews.netfront.net	Cycling, Mountainbiking	154	500	5762	375

Table 5.14: NewsGroup Test Corpuses for Communication Media with Tree-Like Structure.

meaningful the more replies the communication tree contains.
In order to **systematically evaluate the outcome of the clustering algorithms**, the following procedure can be implemented. First, an "objective" test counterpart for the similarity measure is necessary. In order to achieve that, we propose to construct a sociogram by asking any of the members participating in a communication tree to state the strength of their communication relation with the other members of the tree on a standardized interval scale. These data could be compared with the similarity matrix. We could then argue in favor of the performance of the relational clustering algorithms on these similarity matrices (as in the case of the interests) by observing the performance of the relational clustering algorithms on artificial test data.
Unfortunately, these investigations are too costly to be performed within the scope of this thesis.

5.4 Amalgamating Aspect Group-Models

What we have discussed so far are techniques to obtain **meaningful models for groups** (Ad-Hoc-Groups and abstract groups) with respect to three different profile elements. We discussed how to obtain models for Ad-Hoc-Groups and abstract groups on the basis of **highly context sensitive parameters**. Our example was locations & velocities. And we discussed how to obtain models for groups on the basis of **explicitly won profile elements** (our example were interests) and **implicitly won profile elements** (our example were contributions in communication structures with tree like structure. If an application needs a group model with respect to a specific aspect linked with a single profile element, then the introduced group detection techniques can be used directly. If an application needs a group model with respect to a combination of several aspects, the models from the single aspects need to be amalgamated into a **combined model**.

The **group models** of the several aspects consist of a row of a fuzzy or crisp **membership matrix** U_{ik} (Fuzzy: $U_{ik} \in [0,1]$; Crisp: $U_{ik} \in \{0,1\}$) and several prototypical properties of the group which we call **prototypes**.
In case of **conventional fuzzy clustering** (e.g. FCM) or **crisp clustering** (e.g. K-Means), the pattern space is typically a vector space so the prototype vector has a clear direct interpretation

in that space. E.g. in case of location velocity, the prototype vector states the group's position and velocity. This is also true for the computed abstract groups wrt. Locations and velocities.
In case of **RACE based relational clustering** (e.g. RACE-SA) the interpretation of the prototype is also obvious: The group $\mathcal{G}_i$ is represented by a single prototype pattern x_{k_i}. If we want to formalize this as a prototype vector we can do so by setting $(\pi_i)_k = 1$ for $k = k_i$ and $(\pi_i)_k = 0$ for $k \neq k_i$.
In case of **RFAO**, the prototype vector π_i of a group $\mathcal{G}_i$ with $(\pi_i)_k \in [0,1]$ is less obvious to interpret. We can view the components of the prototype vector as the share that a pattern contributes to the pattern that characterizes the whole group.

If we denote the values of the p attributes $A_1, A_2, \ldots, A_p$ of a person's personal profile $\mathscr{P}_k$ as ${}^{(1)}(x_k), {}^{(2)}(x_k), \ldots, {}^{(p)}(x_k)$, where each attribute-value may be an $m(A_j)$ dimensional vector we have several possibilities to compute a **group profile** $\mathscr{G}_i$:

- Determine group structures with respect to a single aspect A_j (e.g. A_j = location & velocity $\Rightarrow {}^{(1)}(x_k) \in \mathcal{L} \subseteq \mathbb{R}^4 \Rightarrow m(A_j) = 4$) and use the prototype vector π_i as the only characterizing element of the group. In this case the only attribute in the group's profile is the aspect A_j with value π_i. This is the case with locations and velocities, where π_i determines the group's average location and velocity.

- Determine group structures with respect to a single attribute A_j (e.g. A_j = location & velocity) and use the membership degrees U_{ik} as weights to characterize the group $\mathcal{G}_i$ from the individual's profiles: $\mathscr{G}_i = \frac{1}{\sum_i U_{ik}} \sum_i U_{ik} \mathscr{P}_k$. In this case it is assumed that the group structure with respect to a single attribute is also reflected with respect to the other attributes. For attributes which have values in a metric space this corresponds to equations (5.6) and (5.63). For attributes which do not have values in a metric space the "sum" needs to be interpreted appropriately (E.g. in case of the tree like communication behavior partly as a weighted sum of the word vectors.)

- Determine group structures with respect to several attributes and combine the results with fuzzy operators like fuzzy union, intersection or complement.

With respect to this last point, many variants of these elementary operations exist in fuzzy set theory [200]. Among the so called t-norms, which are a general class of intersection type set-operators, the min-norm is the most common fuzzy analogon to the usual set theoretic intersection and among the so called t-co-norms we have a fuzzy equivalent for the union operation. Together with the fuzzy equivalent of complementation we have:
Let $\mathcal{A}_1, \mathcal{A}_2$ be fuzzy sets $\mathcal{A}_i = \{(x, \mu_{\mathcal{A}_i}) | x \in \mathcal{X}; \mu : \mathcal{X} \to [0,1]\}$ in $\mathcal{X}$, then [200]:

$$\mathcal{B} = \mathcal{A}_1 \cap \mathcal{A}_2 \quad \Rightarrow \quad \mu_{\mathcal{B}}(x) = \min(\mu_{\mathcal{A}_1}(x), \mu_{\mathcal{A}_2}(x)) \tag{5.89}$$

$$\mathcal{B} = \mathcal{A}_1 \cup \mathcal{A}_2 \quad \Rightarrow \quad \mu_{\mathcal{B}}(x) = \max(\mu_{\mathcal{A}_1}(x), \mu_{\mathcal{A}_2}(x)) \tag{5.90}$$

$$\mathcal{B} = \mathcal{A}^c = \mathcal{X} \setminus \mathcal{A} \quad \Rightarrow \quad \mu_{\mathcal{B}}(x) = 1 - \mu_{\mathcal{A}}(x) \tag{5.91}$$

With the help of these elementary operations, deliberate combinations of clusters can be computed, where the union operation corresponds to logical OR, the intersection operation corresponds to logical AND and the complementation operation corresponds to logical NOT.

Summary

In this chapter basic strategies for the detection and the modeling of Ad-Hoc-Groups and abstract groups were developed. With respect to Location and Velocities, crisp clustering algorithms are well suited tools. The choice of the precise algorithm is not a big concern but the cluster validation and selection procedure is crucial. We developed a socially motivated cluster selection method and investigated the optimal parameter settings. The method performed as expected and the quantitative measures showed an excellent overall performance of the detection and modeling approach. Furthermore, we were able to detect most of the underlying abstract groups by comparing the member structure and the periodicty of the associated Ad-Hoc-Groups. For the detection of groups with respect to interests and communication data we needed to modify the existing approach of relational clustering with the help of Simulated Annealing and through the unconventional approach of Relational Fuzzy Alternating Optimization. On artifical test data, the methods worked substantially better than the RACE algorithm from literature. On our interest test data, the algorithm did not perform very well, although the results of the pre-investigations of chapter 4 showed that the similarity measures should work. This will have to be subject of further investigations.

Chapter 6

Applications for Group Models

In this chapter we will take a look at possible applications for group models. The first application deals with indicating groups. We introduce visulizations and textual indications and explain why the indication of groups could be an added value. Second is a group-model based variant of Collaborative Filtering. After a short review of conventional Collaborative Filtering it is discussed what benefits the usage of group models can bring for Collaborative Filtering. The third application deals with the use of group models for location based information push. Several arguments are presented that speak in favor of using group models for this type of application. The last idea for an application that is closely related to the first and third type is a concept for a location sensitive group reminder and group calendar application

In the previous chapter we have seen how abstract groups and Ad-Hoc-Groups can be detected and modeled. We will now discuss how these models can be used in community support applications for mobile communities.
In chapter 1 we have discussed several general requirements for the support of communities in general and mobile communities in particular. It was discussed that communities are characterized by a common pursuit which we identify with the build-up of a Collaborative Information- and Knowledge Space (CIKS). We classified all actions in a community as communication acts that may change the CIKS. Community support was therefore mainly characterized as support for direct and indirect communication. In a mobile community, services are much deeper integrated into the everyday life of the user. In order to deliver an added value, it was stated in chapter 1 that community support services should be context-sensitive and should support communication in a context-sensitive way. Examples are services that **open communication channels** with respect to location, services that **manage communication channels** according to context parameters (e.g. reachability management) or services that **prepare communication** through communicating (e.g. visualizing) profile parameters such as the location of other users.
Taking these considerations into account, we will now discuss how group models can be used to create new services or to enhance existing services for the support of mobile communities.

6.1 Indicating Groups

The most obvious application of the analysis of abstract groups and Ad-Hoc-Groups is **indicating the analysis results**. It is clear that the models need to be appropriately processed before

the group structures are indicated to offer the user an added value. In general, indication of group structures should be combined with other community services by e.g. offering a suitable communication channel to the respective group that is indicated.
The simplest indication service is to indicate group structures with respect to a single attribute.

6.1.1 Indicating Groups wRt. Location and Velocity

If the attribute is location & velocity, several aspects need to be regarded.
The human visual processing system is a very efficient clusterer whose visual clustering abilities far exceed any computer algorithm however efficient it may be. It does therefore not seem to be appropriate at first glance to e.g. visualize the results of our group detection and modeling process wrt. locations and velocities on a 2 dimensional map, because it seems not to deliver an added value. One could simply visualize the individual user's positions and **let the eye do the job**. Several arguments speak **against this critique**:

- If the number of mobile community members is very large, the eye can well identify groups but the **relevance of the groups** for the single user cannot be easily determined visually, because a large number of points on a map can only be visualized with the help of legends (e.g. the users are visualized with numbered dots and the association of the numbers with the user-ids must be looked up in the legend). If only members of **explicitly declared personal groups** (e.g. Buddylists) are displayed on the Map, this problem does not occur but the user will **miss groups** that are not formed by his buddies or that have only one buddy as a member. Thus such a restricted indication service does not facilitate the build up of new social relations but is only a potentially useful bookkeeping and awareness functionality for existing ones.

- Especially when Ad-Hoc-Groups with respect to context parameters and abstract groups with respect to less dynamical parameters (e.g. interests) are **combined** (see section 5.4), the aspect of relevance can be filtered much more effectively.

- In order to display a sufficiently large number of user-locations for an appropriate overview, a map with a sufficiently **large scale** needs to be used to visualize a sufficient number of user-locations. On this scale, **individual users** in socially relevant groups (see chapter 2) are usually **not distinguishable** any more because the symbols are printed on top of each other. Figure 6.1 clarifies the problem. If groups are modeled explicitly, expandable group symbols could be used instead of printing the individual users on top of each other.

- Furthermore, techniques like the Social Cluster Validation and Selection procedure **SCVS** (see 5.1.3.3) allow for a degree of **expressiveness of the models** (with respect to general relevance and validity of clusters as groups) that is not accessible with the "naked" eye.

- Without detecting and modeling Ad-Hoc-Groups, no **abstract groups with respect to context parameters** can be found and thus no such abstract groups can be indicated. In order to give an overview of a social situation, the indication of abstract groups (e.g. their meeting locations and their meeting frequency) is of great value.

Since groups are so important as social ordering entities, it is therefore a good idea to implement a service that solely visualizes groups and not isolated individual users because the aforementioned

Figure 6.1: Left subfigure: If only individuals are indicated on a 2-d map, the eye identifies clusters which are no socially relevant groups. E.g. the scale of the maps shown equals 1000 m in reality so that e.g. individuals 1 and 3 are over 100 meters apart and individuals 2 and 3 are separated by a large building area. Right subfigure: If the individuals 1,2,3 and 4 are in an Ad-Hoc-Group interaction, their symbols are printed on top of each other.

arguments suggest that such a **distinct group visualization** gives a better overview of the group structure in the respective context (e.g. spatio-temporal situation).

Having a model for a group (which is on the lowest level equal to actually knowing that a set of people form a group) makes it easier to construct visualizations for groups that e.g. provide for **self-expanding legends** which give more social overview with less user interaction compared to expendable legends for every individual. Furthermore, the way in which this "group-indication-service" is combined with other (mobile) community support services (e.g. direct communication with the group) is important for the degree of usefulness of this service. (see figure 6.2 for a suggestion how this might be realized). Of course, internal geo-formats for locations need to be translated into human readable form when handled as text. Usually addresses serve this purpose very well. For an efficient visualization, the length (norm) of the speed vector can be classified according to categories of human motion (walking, driving in a city, driving on a highway, flying etc) and the direction can be classified along main axes of orientation (north, northeast, east, etc.).

Another argument that will appear more often in this chapter, supports the idea of explicitly focusing on the visualization of groups: Group interaction has a normative character: people usually come together because they want to do something specific together and abide to certain general or group-specific rules of social interaction. Therefore, it can be concluded that indicating a group with respect to a context parameter implies that the individuals have some **distinct common activity in the contextual situation**. This is in contrast to indicating an isolated individual whose social state of interaction is less clearly determined in a given con-

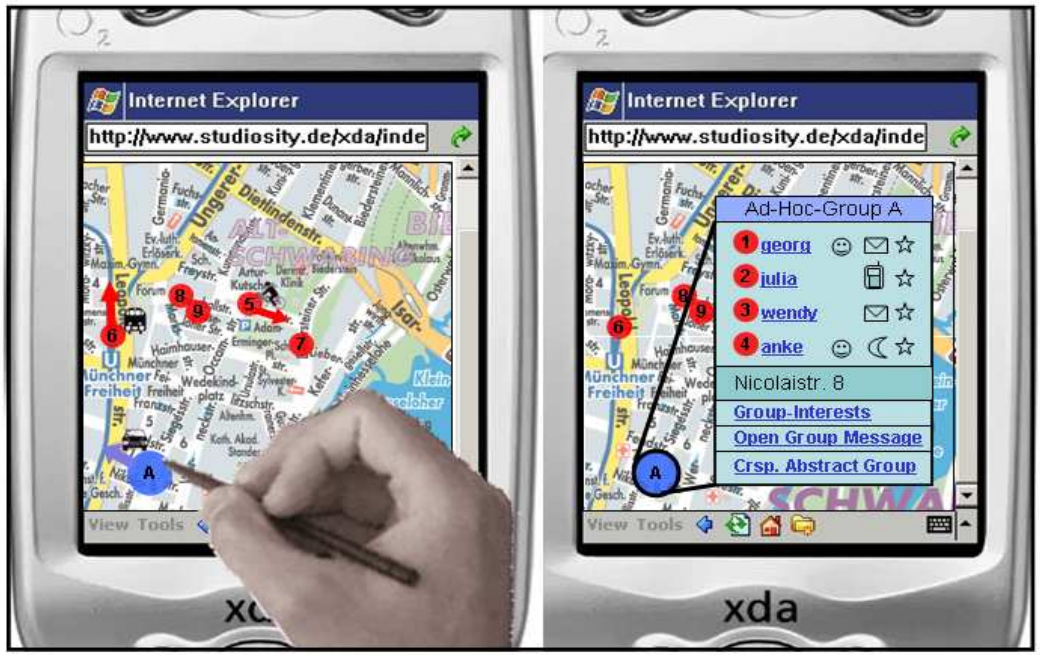


Figure 6.2: Providing indication information on the Ad-Hoc-Group level. By clicking on the group symbol, an information card pops up. Velocities of individuals or groups are depicted with variable length arrows and speed category symbols.

Figure 6.3: Providing indication information on the abstract group level. By clicking on the abstract group link, an information card showing details about the Ad-Hoc-Group's underlying abstract group pops up.

text (The person may be reading, sleeping or talking to people outside the mobile community etc.). Thinking of e.g. possible uses of the group location visualization in contrast to individual location visualization this implies that the **barrier to join a group may be lower**.
Furthermore, the normative character of a group can be an argument with respect to privacy. Since realizing individual context-sensitive access policies for possibly sensitive data such as the location is very difficult, showing only groups and group level access policies could be a way to assure that such information is only available for others in the context of a group interaction. This means that personal location is only available when the probability is high that the person is already in a social interaction with others. The normative character of group interaction ensures that the probability that a possible revelation of a personal location might be embarrassing may thus be smaller.

6.1.2 Indicating Groups wRt. Interests And Communication

In contrast to indicating groups which have been modeled as crisp clusters, interests and tree-based communication induce **fuzzy group models**. Furthermore the group prototypes consist of **texts or text fragments** and need to be preprocessed before being suitable for indication. This makes the indication service much more complicated.

The first alternative in indicating groups with respect to interests and textual communication is the **plain textual representation of detected abstract groups** by simply listing the members. We have modeled the groups as fuzzy sets (membership matrix elements $U_{ik} \in [0,1]$) which means that in principle every user is member of a fuzzy group in question to some degree. In order to list the members of a group in a textual form, one possibility is to perform **defuzzification** e.g. by introducing a threshold for the defuzzification [161]. This means that a fuzzy group is turned into a crisp group by defining a threshold membership value μ and setting

$$x_k \in \mathcal{G}_i \leftrightarrow U_{ik} > \mu \tag{6.1}$$

The disadvantage of this process is that the threshold μ is a crucial element which is hard to determine by heuristics or via the internal structure of the data. Furthermore, information is lost during defuzzyfication.
An alternative is to keep the fuzzy membership degrees and to code them in a way which gives a good overview. A generally applicable way to visualize fuzzy sets is via **color encoding**.

6.1.2.1 Visualizing Fuzzy Sets

Colors can be encoded as RBG vectors on a three-dimensional sphere. It is reasonable to assume that two colors have the highest discriminative power, if the angle between their RGB vectors is maximal. Therefore the problem of choosing c base colors for c clusters (groups) is the problem of finding c points on a three dimensional unit sphere so that the angle between the points is maximal.

The problem is well known in physics (n equally charged particles on a sphere) and mathematics (See [170], [193], [119]). Unfortunately, the exact solution of the problem is still unknown, but various numerical methods exist, that allow for the computation of such points [170]. In our case the problem is slightly modified because the color space is restricted to the segment of the unit sphere in the first octant ($x \geq 0, y \geq 0, z \geq 0$).

If we assume that c three-dimensional RBG vectors $\{(r_1, g_1, b_1), (r_2, g_2, b_2), \ldots, (r_c, g_c, b_c)\}$ have been computed (one for each cluster), we can compute the colors for the n symbols of the users on the map by summing up the weighted base colors of each cluster. The weight of the base color (r_i, g_i, b_i) of cluster $\mathcal{C}_i$ is the membership degree U_{ik} of the user k in that cluster. Thus the color for user k will be:

$$(r^{(k)}, g^{(k)}, b^{(k)}) = \sum_{i=1}^{c} U_{ik} \; (r_i, g_i, b_i) \tag{6.2}$$

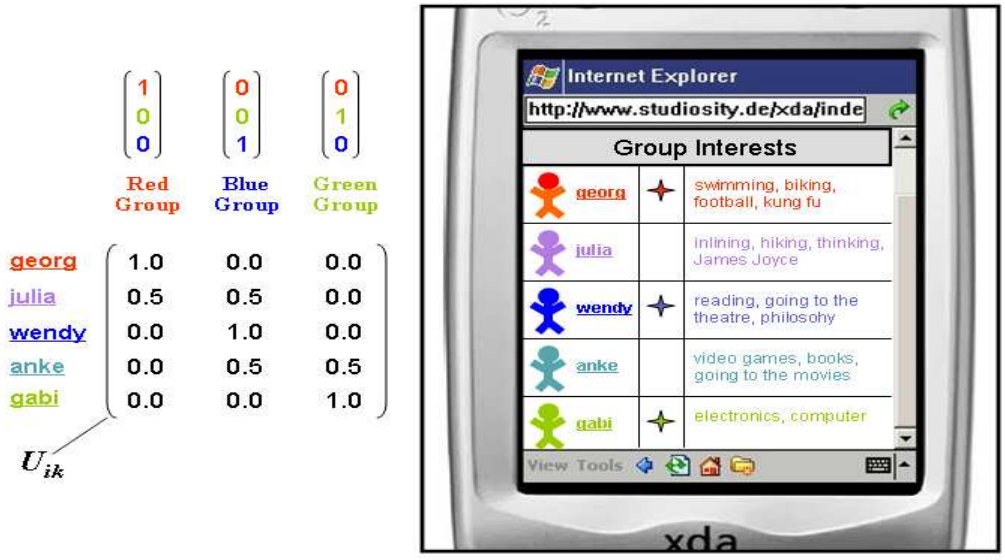


Figure 6.4: Visualizing fuzzy groups with the help of color encoding. Three basic colors are appointed to three groups which are represented by their three prototypes. The other two users are in between group one and two and two and three with respect to their membership degrees. Group prototype users are marked with a star symbol.

6.1.2.2 Processing the Prototypes

In case of the **interests**, we usually have a **unique prototype pattern** that represents the group (RACE-SA based detection and modeling) or we have a prototype pattern vector (RFAO based detection and modeling). In the case of a prototype pattern vector we can determine the element with the largest value and assume this pattern as the unique prototype pattern. The interests of the prototype user representing the group can then be directly displayed for all members of the group (after membership degree based defuzzification). The drawback is that the membership degrees and the interests of the group members are not reflected at all. Thus a more appropriate approach would be to weigh the interests with the membership degree U_{ik} and list the interests of the members in the group in a ranked order where U_{ik} determines the rank. If enough space is available, the interests of the users with the next best membership degrees apart from the prototype user will also be visible. This facilitates a better overview about the

overall interests in the group.

In case of the **communication** groups, the prototypes are more complicated data structures. On the one hand we have for every user k the vector of **reply frequencies** $(m_{(k)\to(k_1)}, m_{(k)\to(k_2)}, \ldots, m_{(k)\to(k_n)})$ to the other user's postings and on the other hand the vector of aggregated **reply contents** $(x_{(k)\to(k_1)}, x_{(k)\to(k_2)}, \ldots, x_{(k)\to(k_n)})$. While the former is mainly important in detecting the groups, the latter is the main aspect of the prototypes which is able to show the prototypical properties of the group. Obviously, we cannot simply list the full contents nor the word vectors of the communication contributions within the group (after e.g. a defuzzification). What is necessary in order to give an overview about the contents of the communication between the persons in a group is a **keyword-** or **key-phrase-extraction** from the posting contents which can give a meaningful overview of the topics of the communication within a group.

Keyword Extraction The field of keyword extraction and key-phrase extraction is a relatively well researched field in information retrieval and related disciplines (see e.g. [112, 45, 75, 113]). Generally we can distinguish between some principle approaches in extracting key-elements from texts: The easiest and most widespread approach uses only **statistical properties of the text** and the other texts from the collection that the text is taken from. More elaborate approaches use extensive **NLP preprocessing** of the text on the syntactical level (with parsers and taggers). even more elaborate are approaches that also use **semantic resources** like Ontologies to determine semantic relations between the words. A detailed description of the various approaches is beyond the scope of this thesis.
The basic idea of the **vector space based approach** (see section 4.8.3.1) to keyword extraction is to define a corpus of documents and to determine $tfidf$ word vectors for them and then to take the most frequent words as keywords. Approaches that use domain-knowledge (e.g. from a community specific ontology) may determine semantic relations between the words in the postings with respect to that domain knowledge, compute clusters on this high dimensional relational matrix and use the cluster whose words have the highest $tfidf$ frequency as keywords. (see [75] for other alternatives). Methods that use more sophisticated NLP techniques could rely on detecting key sentences instead of keywords (see [45]).
For the reasons of simplicity and space restrictions will only present a variant of the most straightforward approach that is based on the vector model that is also basically used in [45, 75, 113]. We will combine this approach with the fuzzy group model in order to compute a meaningful communication prototype for a group.
Applying the vector space idea to the postings that have been made between members of the group can be accomplished by weighing the postings with the fuzzy membership matrix U_{ik} (Always keep in mind that for fuzzy groups the term "group members" needs to be used with the right interpretation in mind):
For user k the aggregated word vector of his answer postings to user k_j was notated as $x_{(k)\to(k_j)}$. If we want to compute the aggregated word vector $x_{(k)\to(\mathcal{G}_i)}$ for user k that takes into account users in group $\mathcal{G}_i$ we can do so by using the membership degrees of the users k_j in group $\mathcal{G}_i$ as weights:

$$x_{(k)\to(\mathcal{G}_i)} = \sum_{k_j \neq k} U_{ik_j} x_{(k)\to(k_j)} \tag{6.3}$$

In order to get a meaningful construction we must use normalized word vectors:

$$||x_{(k)\to(k_j)}|| \stackrel{!}{=} 1 \tag{6.4}$$

Of course the contributions of a user k towards users of the group $\mathcal{G}_i$ are only relevant for the group's communication to the same degree that user k is member of the group $\mathcal{G}_i$. Therefore the "document word vectors" $\check{d}^{\langle k \rangle)}$ building the corpus which characterizes the group $\mathcal{G}_i$ are

$$\check{d}^{\langle k \rangle} = U_{ik}\ x_{(k)\to(\mathcal{G}_i)} \tag{6.5}$$

We can now use e.g. the h words with the highest weight from these documents to characterize the content of the communication within the group $\mathcal{G}_i$ as the set of keywords $\{w_{p_1}^{(\mathcal{G}_i)}, w_{p_2}^{(\mathcal{G}_i)}, \ldots, w_{p_h}^{(\mathcal{G}_i)}\}$.

6.2 Collaborative Filtering Revisited

In the previous section we have seen how the group analysis results (group models) can be used to construct more or less sophisticated group indication services. The main social function of these services can be seen in the field of (group)**awareness** and **matchmaking**.
If we think of using the group models to directly control information flows in the community's CIKS we arrive at a new class of applications. The simplest of such applications is **Collaborative Filtering**. We will now discuss how group modeling can be used to improve the usefulness of Collaborative Filtering. This section will not explicitly apply to mobile communities and context. These aspects will be discussed in the next section.

6.2.1 Conventional Collaborative Filtering

Collaborative Filtering is essentially about **predicting the degree of relevance of an information-item** for one person **on the basis of degrees of relevance** of that information item for **other persons**. The degrees of relevance of the item for other persons is either estimated **implicitly** or raised **explicitly** in form of **ratings**. The relevance degrees for the other persons are generally weighted with the strength of a relation from the person in question to these other persons. It is thus necessary that these other persons can be compared to the person in question via some relational measure.
Denoting the rating of user k for item j by v_{kj} and the set of items that user k has voted on by I_k and the average voting of user k by

$$\overline{v}_k = \sum_{j \in I_k} v_{kj} \tag{6.6}$$

we can **predict** or estimate the voting of user k for an item with index j_m that he has not seen or rated yet by [21]

$$v_{kj_m} = \overline{v}_k + \alpha \sum_{k_a=1}^{n} w_{kk_a}(v_{k_a j_m} - \overline{v}_{k_a}). \tag{6.7}$$

In this general formulation, the weights w_{kk_a} are a measure for the similarity or correlation between users k and k_a and α is a normalization parameter. It is the calculation of these weight

parameters that distinguishes the basic approaches to Collaborative Filtering. The most popular variants of weights are e.g. the Pearson correlation [21]

$$w_{kk_a} = \frac{\sum_j (v_{kj} - \overline{v}_k)(v_{k_a j} - \overline{v}_{k_a})}{\left(\sum_j (v_{kj} - \overline{v}_k)^2 (v_{k_a j} - \overline{v}_{k_a})^2\right)^{\frac{1}{2}}} \tag{6.8}$$

or cosine measure

$$w_{kk_a} = \sum_j \frac{v_{kj}}{(v_{kj}^2)^{\frac{1}{2}}} \frac{v_{k_a j}}{(v_{k_a j}^2)^{\frac{1}{2}}}. \tag{6.9}$$

6.2.2 Improving CF via Group Models

All these ways to calculate the similarity or correlation between users are based solely on the ratings of the users. As has been mentioned before, these ratings can be gathered explicitly (by interacting with the users) or implicitly (by inferring them via measurable parameters such as the time a user views item j or the frequency of interaction etc. (see e.g. [27])). In any case, the similarity is calculated on the basis of **user-item relations** ("ratings") alone and neglects other relations such as **user-user relations, group structures** etc. (see sections 1.3.1 and 1.3.2) that are very useful for calculating the weights.

In the **conventional Collaborative Filtering** approach, we have the **assumption** that the rating for an item from a user k_1 is more important for guessing the usefulness of the item for a user k_2 the more similar the rating behavior of both users is. In that way, the Collaborative Filtering process is assumed to be "self-adjusting": No matter what the other relations between the users are like: If they like the same items, a new comparable or similar item liked by one of them may be of use for the other. As an example consider a movie recommendation system: If a university professor and a Harley-Davidson biker both liked several Walt-Disney-Movies and the Biker liked the latest movie from Disney, the Professor is likely to fancy this movie as well, no matter how dissimilar he may be from the biker with respect to other aspects. But the key point is that the new **item must be similar** to the previously related ones: If the "Disney-trained" Collaborative filter is used to predict the professors affinity to a new Harley-Davidson bike it will probably not produce good results. Thus the idea of Collaborative Filtering only works when the items to be filtered are similar to the items which have been used to train the Collaborative Filtering system which is, in essence, nothing more than a supervised classifier. Thus, if we have very heterogeneous information items the conventional approach does not work well.
The argument can also be turned around: if the **users are similar** to each other and the new item is not very similar to the previously seen ones, then the similarity between the users is a criterion to nevertheless positively recommend the item. However, the similarity aspect between the users should match the "topic" of the item.

So what can be generally assumed is that **Collaborative Filtering works best**, when

- **User-Item-Relations** (ratings) are similar between users.
- **Item-Item-Relations** indicate similarity between filtered items
- **User-User-Relations** indicate similarity between filtering users

While the first aspect is (as has been discussed above) directly respected in the conventional approach represented by equations (6.7), (6.8) and (6.9) the second and third aspects are usually only implicitly respected by restricting the topic focus of the platform that offers CF.

The most obvious idea to include user-user-relations is to **complement the weights** calculated on the basis of rating-similarity with weights calculated on the basis of **user-user similarity**. The user-user similarity should be calculated with respect to the general topic bias of the items in question. For example, if we aim at filtering political news articles, it might be of interest to add weight components that reflect the similarity with respect to membership in political organizations.
As an example for such an augmentation, consider a **community** with n members, identified by an index k, sharing a Community Information and Knowledge Space CIKS and each member has stated a set of free text interest phrases $\mathcal{X}_k$ as in 4.1. We could then estimate user-user relations with respect to these interests with the help of the similarity measure $\mathrm{sim}(\mathcal{X}_{k_1}, \mathcal{X}_{k_2})$ developed in 4.1 (shown in figures 4.9 and 4.10) [1] The user-item-relation based **weights** from equations (6.8) and (6.9) could then be **complemented as**:

$$w'_{kk_a} = \frac{1}{1+\beta}\big(w_{kk_a} + \beta\ \mathrm{sim}(\mathcal{X}_k, \mathcal{X}_{k_a})\big) \tag{6.10}$$

where β is an arbitrary mixing parameter.
But what are prerequisites for calculating such user-user-relations? Regarding **arbitrary users** on the internet, it is often hard to **compare them** with respect to an aspect because they will in general not have an easily comparable profile from the **technical point of view**. Thus when implementing a Collaborative Filtering service that is based on or has contributions from similarity measures between users, the users should have a comparable profile as is the case in a **community**. From a technical point of view, **item-item-relations** can also be much easier determined if the items are from a community's CIKS.
From a **social point of view**, the **topical homogeneity** of the filtered **items** is more likely if they come from a community's CIKS. A further argument for community being a good prerequisite for including user-user similarities in the Collaborative Filtering weights are the **social bonds** that usually exist **between community members**.
In the suggestion (6.10) these social bonds represented by a membership in a community are not reflected: no matter if users are members of the same community, if they have similar interests (user-user-relation) and have similar rating behavior (user-item-relations) their mutual recommendation for a new item is accepted.
From an **abstract modeling point of view**, the affinity of a user towards a community should thus be reflected by introducing a membership selector $U_k \in \{0, 1\}$ (1 if user k is member of the community, 0, if not) that restricts the social basis for Collaborative Filtering to the community

$$w'_{kk_a} = \frac{1}{1+\beta}\big(w_{kk_a} + \beta\ U_k U_{k_a} \mathrm{sim}(\mathcal{X}_k, \mathcal{X}_{k_a})\big) \tag{6.11}$$

Taking the idea of adding **user-user-commonalities** such as the common membership in a community serious means that we should look for more suitable social structures which have a high probability of being reflected in algorithmically accessible data. **Ad-Hoc-Groups and**

[1] Regard that in chapter 4 sets of free text interest phrases have been indexed with an upper index j because in this chapter the development of a general similarity measure on a set of such sets of interests was the focus and the authorship was of less interest.

abstract groups are natural candidates for such structures on the sub-community level. While we will investigate the possible uses of Ad-Hoc-Groups in the next section, we will now argue why **abstract groups** are natural weighting aspects for CF to be regarded.
Where a community's common pursuit would be too general and where the mere organizational aspect of a community (as a provider of service bundles) is very prominent, abstract groups within the community are the **"self-adjusting"** component for CF from the user-user-relation perspective. Groups play a tremendously important role in structuring the personal knowledge and information sphere (see 2) and it is reasonable to assume that within an abstract group

- users have **tight social relations** with one another. Example: Family
- a **group-CIKS** can be abstractly associated with the group which is likely to be **topically focused** on the group's main social **coherence aspect**. Example: A group of teen girlfriends having a huge archive of George Clooney pictures and articles.
- if the items in this group-CIKS are **not directly topically related** (trivial relation), they are likely to be **related indirectly** via the **social brace** that the group represents. Example: group of several retired old chess players may have documents about chess opening strategies as well as documents about old red wine and expensive cigars.

Especially if two or more aspects are combined in the course of the detection phase, the group is likely to have a real existence (e.g. abstract groups wrt. location and velocity and abstract groups wrt. interests are intersected) and thus is likely to be a natural basic population for knowledge management and information flow control services such as CF.

We denote (as usually) the membership degree of a user k in a group $\mathcal{G}_i$ (which can be a combined group (via the methods described in section 5.4) or a single aspect group) by U_{ik}, the group's prototype by π_i and the item that is to be filtered by I_j. There are various possibilities to include the group model into the augmentation / complementation equation for the weights which replaces and generalizes the rather artificial community membership selectors of equation 6.11.

$$w'_{kk_a} = \frac{1}{1+\beta}\big(w_{kk_a} + \beta \;\; U_{ik}U_{ik_a}\mathrm{sim}(k,k_a)\big) \tag{6.12}$$

$$w'_{kk_a} = \frac{1}{1+\beta}\big(w_{kk_a} + \beta \sum_i U_{ik}U_{ik_a}\mathrm{sim}(k,k_a)\big) \tag{6.13}$$

$$w'_{kk_a}(I_j) = \frac{1}{1+\beta}\big(w_{kk_a} + \beta \sum_i \mathrm{sim}(\pi_i, I_j)\; U_{ik}U_{ik_a}\mathrm{sim}(k,k_a)\big) \tag{6.14}$$

Equation 6.12 uses only one group membership to filter the general user-user similarity $\mathrm{sim}(k,k_i)$ (for which we can set e.g. the interest similarity $\mathrm{sim}(\mathcal{X}_k, \mathcal{X}_{k_a})$). Equation 6.13 uses all existing groups to filter the general user-user similarity and equation 6.14 introduces a further similarity which aims at comparing the group's "topic" with the item to be filtered. This similarity measure would have to be defined e.g. on the basis of the vector model or more sophisticated means.

6.3 (Proactive) Context-Sensitive Information Push

It is the principle of all classes of all so called **proactive information services** to analyze aspects of user interaction with an information system in the past to predict information needs that a user might have in the future. The methods vary from tracking the web-sites a user visits to analyzing his eye movements, the models induced are various and the time scales on which the user interests are extrapolated vary from minutes to weeks.

All these approaches have **drawbacks** in the accuracy of their predictions because of the rapid change in context and thus in the rapid change of information needs of human beings which is very hard to predict. Context-sensitive systems try to improve prediction accuracy by measuring more fine grained context-parameters and by building more fine grained models of user interests.

It can be argued that the information needs of a single human being change more rapidly than the information needs of users in the social context of a group because the **social context of a group** tends to **focus the information needs**. The information needs are correlated with the group's characteristics with a high probability. Furthermore, the members of a group contribute more context data for the characterization of the group than a single user contributes for characterizing him only.

As an **example** consider a single user sitting in his office. You would have to have a very fine grained model of his context to predict whether he will in a minute need just some deflection in form of a comic strip or whether he will need some information on the project he is a member of. In contrast to that, a group of soccer enthusiasts in football stadium will appreciate information on the scores of the other games with high probability because social context focuses information needs. Of course, some soccer enthusiast might as well be interested in ancient philosophy but if he is with many others who are interested in soccer the probability of an information need in the field of soccer is much higher.

In order to work out the benefits that an integration of group models and especially of Ad-Hoc-Group-Models can provide for proactive context-sensitive information services we will focus on an obvious example: location based information services.

6.3.1 Location Based Information Retrieval and Filtering

Thinking about proactive information services that are sensitive to the current context of the user, the most obvious context is represented by the user's **location** and thus the corresponding service is about finding and providing **information that is relevant with respect to location**.

6.3.1.1 The Problem

Information retrieval is often distinguished from **Data Retrieval**. The latter is concerned with finding data in **formally structured data sources** and the former is concerned with finding data in **semi-structured data sources**. The typical examples for these two extreme cases are Databases and Information Retrieval in natural language texts. **Databases** are typically formally structured according to a Relational Algebra and formal languages (like SQL) are used to define what elements from the source data structure are to be retrieved (the formal information need). The user expects to get full Recall and Precision from such a system (all the

relevant data and only the relevant data are retrieved). **Information retrieval in natural language text corpora** uses semiformal models to represent the structure and semantics of the given texts (Vector Model etc.) and the query is also formulated in natural language (e.g. with keywords). The results are generated by a heuristic comparison of the query with the models of the corpus (e.g. Google's Page Rank) and Precision and Recall are typically less then one.

If we transfer this to the problem of **finding data with respect to geometric aspects** we can also identify the two extremes (see [104]). On the one hand, we have **specialized databases** such as Oracle Spatial [125] and formally geo-coded data (e.g. electronic maps with points of interest) which are a key market asset in the field of location based services and are marketed by companies such as TeleAtlas [124]. These data are encoded in special formats like GDF (see [124]) which can be imported into spatial databases (like Oracle Spatial) or car-navigation units etc.. A spatial database allows for formalized queries which specify information needs related to points, polygons, areas near or adjacent to other spatial structures etc..
On the other hand, we have systems that work on **semi-formal natural language documents** and are able to process informally stated natural language spatial queries such as "where is the nearest restaurant" by crawling these semiformal documents (like e.g. web pages) for formulations that reveal spatial relations and dependences and matching those with the result of analyzing the semi-formal query with NLP techniques.
We implemented a very **simple proof-of-concept-system** that was able to crawl the web for postal addresses (the most common human readable geo-coding format) in web pages that were concerned with restaurants, bars and other predefined types of places of local interest. Instead of a natural language query system, we implemented a simple boolean query system which was able to express information needs such as "restaurants or bars but not fast food". The system was equipped with a set of simple heuristics that allowed for the recognition of German postal addresses and the association of the page with a restaurant or other point of interest of predefined type. A mobile web-based interface was established on an XDA (see section 1.5) and connected to the COSMOS location tracking and address resolution system. The user was then able to access links to web pages which matched addresses and specified point-of-interest types in his immediate neighborhood (see [91] for details).
While **geo-coded data retrieval** is already a well developed field, spatial or **geo-referenced information retrieval** is still in its infancy. The main reason why this is nevertheless an important field of study is that the **data retrieval approach** has one important **disadvantage**: the data needs to be **manually input and maintained**. Thus we can call formal data sets **restricted information spaces**. In contrast to that, semi-structured information like web-page-contents is more fuzzy but at the same time able to constitute **free information spaces**: information can be maintained freely in a distributed way.
In the field of **general data-retrieval vs. general information retrieval**, **initiatives** have been started in the last 10 years to move both worlds closer together and to allow for data-retrieval methods to be incorporated into information retrieval. The most famous initiative is the **Semantic Web** [10]. It aims at enriching semi-structured information sources with meta-data which declare structural and semantic aspects of these sources. Main semantic declaration instrument are knowledge representation languages such as OWL [67] which are Description Logic based (see [42]) and allow for database-like queries and reasoning on the meta-data.
In the special field of **geospatial data retrieval** vs geospatial information retrieval such efforts are on the way too. GML [150] is a multipurpose meta-data language that e.g. allows for adding geospatial metadata to information sources which can then be queried in various ways. Other

projects such as SPIRIT [80, 68] use the Semantic Web standards to develop special ontologies for geospatial information retrieval and meta-data annotation that allow for the specification of facts such as "this information source is relevant along a line which is defined by..." (e.g. as metadata for documents which are related to a highway) or "this information source is relevant in a polygon specified by ..." (e.g. as metadata for documents about a city neighborhood) etc.. (see [171]). Within this project, other groups are concerned with NLP aspects of geospatial information retrieval such as the extraction of information linked with location relevant formulations such as "near to", "south of" etc. [68].

6.3.1.2 Location Based Information Retrieval and Filtering in Mobile Communities

In a mobile community, location based information retrieval and filtering have a special relevance. As we have seen in chapter 1, **general mobile interaction with information spaces** strongly profits from context-sensitivity and especially from **context-sensitivity with respect to location**. Numerous projects aim at investigating the issues related to this goal (context-sensitive tour guides, personal navigation systems etc.). location aware information services take the user's location automatically into consideration without the need to for the user to explicitly code his current location during the access to these services. A much cited example are services that aim at information needs represented by questions like "where is the next Chinese restaurant that offers Dim Sum?". As has been explained before there are also scientific projects that aim at investigating the "where" in this question more thoroughly ("where = in a circle around my position with radius", "where = along the road that I am currently driving on" etc.). On the **scale of the web** it will be a **long way** until all information items that wish to declare a relevance with respect to a geometrical modeling primitive (a direction, a polygon etc.) can do this with the help of **standardized**, appropriate semantic constructs using e.g. a standard ontology for such geometric primitives. If we do not rely on the Semantic Web approach which is somehow nearer to the Data Retrieval paradigm (declarative and formal) but put more faith in **NLP-based resolution** of location primitives in the queries and the space of semi-structured natural language information items we will probably have to wait **even longer** since this is a **very** complicated task.

As has been explained before, these two strategies are possibilities to tackle the problem of location aware information services in the case of **free information spaces** that can be freely altered or supplemented by anybody. In the case of **restricted information spaces** (data retrieval case) all these efforts do not necessarily have to be taken since any company offering geo-coded information (e.g. geo-coded points of interest) can develop a proprietary solution to the problem of interrelating the location reference of the query (or the context of the user) with the location reference of the data.

In a mobile community, we have a big advantage over the web as a whole because we do not have to adhere to global standards for the declaration of location relevance of information items. Furthermore we also do not have the disadvantage of the **restricted information spaces** case, because the community's CIKS is a **free information space** which can be maintained freely by the members and is not bound to e.g. the CD of a car navigation system. For example in a community as COSMOS Studiosity, a lean XML tag language could have been developed which would have allowed to tag the information items with the Gauss-Krüger Coordinates of a location. The mobile user interface which allowed the input of new information items could then be complemented by a simple UI-element which adds a relevance tag wrt. the current

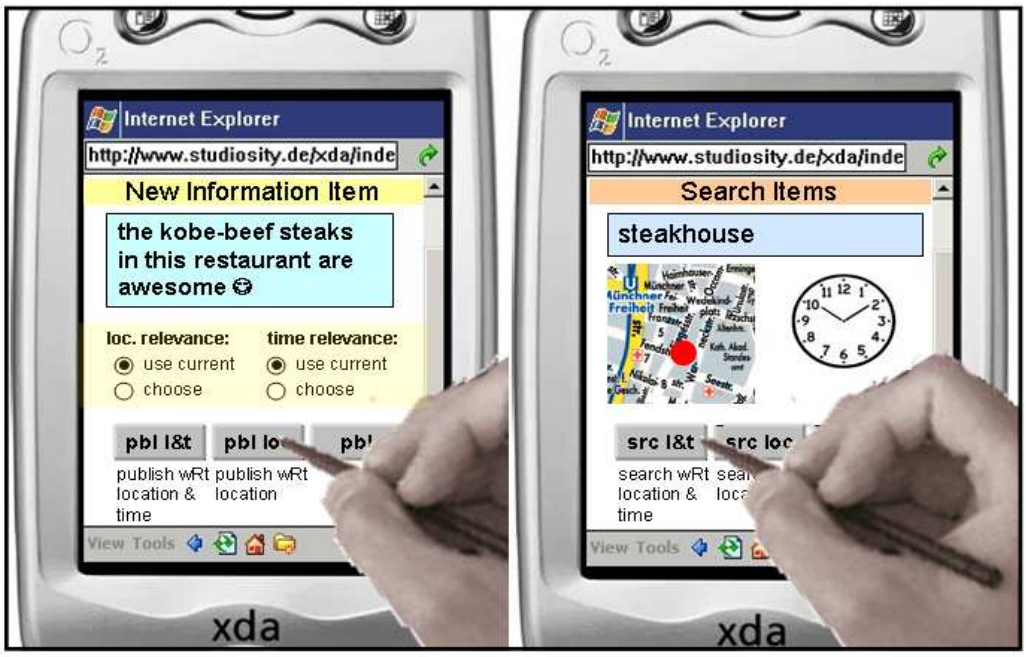


Figure 6.5: Input and retrieval of an information item with location relevance.

location to the item (see figure 6.5). This would in essence be an anonymous 1:m variant of the "virtual post-it" (location based messaging) service introduced in section 1.5 if a respective location based querying of the set of tagged information items was introduced. A proactive variant of this location based information retrieval service in the community's CIKS would sent location relevant information to the user's mobile device in an unobtrousive way (**location based filtering**).

6.3.1.3 Using Group Models for Location Based Filtering

So far we have mostly been talking about single user interaction with information services. In a **community** we have several more possibilities because we can compute models about social structures within the community and support the related **social processes** through community services. With respect to location based information filtering, the **usage of group models** can be an **improvement compared to using single user models**.
Information filtering usually works by assuming a more or less **general, ongoing information need** on behalf of the user. This information need is reflected in a **user model or user profile** which is computed heuristically (e.g. by monitoring the web-sites that a user has viewed). The heuristics should be targeted towards **reflecting the user's information needs**. The evolving set of information items from some information space is then continuously matched against this profile and new or altered items are presented to the user in an appropriate fashion. The main advantage of such a proactive information filtering service compared to an information retrieval service is that it can satisfy **implicit information needs**. In that sense it can be compared to advertising or customer relationship measures (CRM) in Marketing where a former and / or potentially new customer is addressed with a constant stream of information that he

never explicitly demanded. If the Marketing measures are thoroughly planned and the target group for the measures is carefully chosen, a win-win situation for the company and the user occurs. In that sense, the user may profit from information that he never demanded but which is nevertheless useful for him.
The **main disadvantage** also becomes apparent when comparing information filtering to marketing measures: **Spam**. The possible information needs of a single user change so rapidly and are so difficult to predict that the majority of information items presented or the majority of advertising or CRM items is irrelevant, unwanted and perceived as spam by the user.
Using Ad-Hoc-Groups and abstract groups as targets for information filtering and especially for location based information filtering and using Ad-Hoc-Group models and abstract group models as models for representing the ongoing information need has several **beneficial effects**.
The most obvious effect of using a group's model compared to using a single user's model is that we have much **more data** at hand that allows us to calculate a model which really reflects the information needs of the group members. Consider an **example**: A football team meets twice a week for practice and once every second Saturday to watch their team live in the football stadium. The team will thus be identified by the means introduced in the previous chapters as an abstract group because of their interests or as an Ad-Hoc-Group (and thus also as an abstract group) during and due to their regular meetings. Interest in football can be considered their current common denominator. While ratings for literature or music might cancel each other out or deliver only a slight bias towards these topics, the football interest will in general be strongly reflected in the group's model. More generally spoken, the **group's generative element** (football in the case of the example) is likely to be **amplified** in the group model because it is reflected in every group member's individual profile.
Another important effect of using a group's model compared to using a single user's model is that **groups focus the information need** in the course of group interaction (see chapter 2). What does that mean? Consider a single user as a target for a proactive information filtering service. When the single user's football interest is substantially reflected in the data available for profile calculation, an information filtering system may consequently decide to push the latest football results to that user. But the system has no means of knowing when the user is in the right context for sending him this information item. He may be at a funeral or in some other inappropriate situation. In contrast to that, if a person is in an Ad-Hoc-Group, his information needs are focused by the group. It would be considered inappropriate by his football friends if a user would read philosophical literature during their Saturday stadium visit. Thus if the filtering system pushes the information related to a group's generative element (the common denominator) during the group's instantiation as an Ad-Hoc-Group the chances that the information item will match information needs of the users in the group are much higher than in the case of a single user. With respect to location based information retrieval we also have a much better chance of delivering relevant information if we push the item in a context (location and time) where Ad-Hoc-Group interaction occurs. A last but very important aspect that speaks especially in favor of the usage of combinations of Ad-Hoc-Group models and correlated abstract group models is that the **groups** that are detected and modeled with respect to **highly dynamical context parameters** especially when occurring periodically have a higher probability to be **real social structures** than groups that are detected on the basis of conventional community data only (see chapter 2). As an example consider a community with a large number of members (such as jetzt.de [131]). If we apply the procedures of chapter 4 for the detection and modeling of abstract groups wrt. interests the results may or may not reflect real social structures

(which does not necessarily diminish their value for information filtering or group-visualization) Detected Ad-Hoc-Groups wrt. location and velocity especially periodic groups will have a substantially higher probability of representing real social structures. If Ad-Hoc-Group models and models with respect to less dynamic parameters are combined, the results should provide an even higher probability and expressiveness with respect to reflecting real social groupings.

6.3.1.4 An Example

As an example a service can be constructed that uses the models that have been developed in the previous chapters in a rather simple way. The results of clustering based on a dynamically changing profile parameter such as location $(U^{\text{loc}}, \Pi^{\text{loc}})$ and of clustering based on a slowly changing profile parameter such as interests $(U^{\text{intr}}, \Pi^{\text{intr}})$ can be combined by determinig for all groups G_i^{loc} in U^{loc} the group G_j^{intr} in U^{intr} which is most similar with respect to membership degrees. This similarity can be determined using a simple cosine measure:

$$\text{sim}(G_i^{\text{loc}}, G_j^{\text{intr}}) = \sum_{k=0}^{n} U_{ik}^{\text{loc}} U_{jk}^{\text{intr}} \tag{6.15}$$

(see section 5.1.4.1 for a thorough discussion of comparing **crisp** groups with respect to their members).
A threshold should be established that rejects finding pairs of clusters which have only a marginal overlap.

From there we have several possibilities:

- Sending all information that matches π_j^{intr} to all users in the crisp group represented by $\mathbf{D}_c(U_i^{\text{loc}})$ (where $\mathbf{D}_c$ denotes the threshold de-fuzzification operator using a threshold of c (we can chose $c = 1/|\mathcal{X}| \sum_{k=1}^{|\mathcal{X}|} U_{ik}^{\text{loc}}$ as the arithmetic mean of the membership degrees)). Comparing π_j^{intr} with documents or information items from the community information space can in the most simple case also be done by cosine similarity (3.29). This assumes that long term interests will reflect the information needs of the Ad-Hoc-Group $\mathbf{D}_c(U_i^{\text{loc}})$. Furthermore, the degree of urgence that the contents in question would be presented with can be weighted with the membership degrees in the fuzzy group. The degree of urgence of a message or a content presentation determines the way in which these items are delivered to the user. E.g. for a more "urgent" message faster communication channels could be chosen or the volume of the indication signal could be increased etc..

- Sending all information that matches π_i^{loc} to all users in the crisp group represented by $\mathbf{D}_c(U_j^{\text{intr}})$. This variant assumes that information matching the dynamic location information reflects the information needs of the interest cluster. In order to compute relevance of information items with respect to π_i^{loc}, location information can e.g. be reverse geo-coded which means translated into address-range-strings which then can be compared to possible address occurences in the items by cosine similarity or boolean similarity.

- Sending all information that matches π_i^{loc} and π_j^{intr} to all users in the crisp group represented by $\mathbf{D}_c(U_j^{\text{intr}})$, $\mathbf{D}_c(U_i^{\text{loc}})$ or $\mathbf{D}_c(U_j^{\text{intr}} \cap U_i^{\text{loc}})$ (where $\cap$ denotes the fuzzy intersection operator).

6.4 Group Calendar / Reminder

Another interesting application that is related to Location Based Information push and Group Indication is **location based reminder services**. The basic idea of a location based reminder service is the following: If a user keeps an up-to-date calendar, usual PDA's have automatic reminder functions that become active in a configurable time interval before a date and issue a reminder signal. The same idea is transferable to other context parameters such as location: If each date is marked with its location, a mobile system can remind the user if he is in the neighborhood of the date's location. Combinations of time and location are also imaginable. Such a function can make Traveling Salesman types of planning actions obsolete, if its foreseeable that route changes are frequent and optimal planning is costly. A simple application could be logistics for dynamic e-commerce where the products or services need to be delivered as soon as the electronic order is received. The server would then update the calendar of the most appropriate mobile company member and the member would be alerted whenever he is near the location that the goods or services need to be delivered to or performed at. The service is especially interesing, when the target nodes are also mobile and move. A further refinement of the service could be constructed if not only the locations are matched but also means to reach the locations ("You are just near a subway station that will take you to date XYZ").

Ad-Hoc-Group models can be of interest for this type of application as well. In the aforementioned scenario, only two parties at a time need to be monitored: The "deliverer" and the "customer". In case a group of people has a common calendar, common interest or common task that is flexibly scheduled in some type of "fuzzy" calendar, the formation of the group and thus the fulfillment of the task can be assisted by reminding any user that the group in question is forming (some have already joined the group some others haven't) if he is near enough to join. A fuzzy calendar could be constructed with the help of fuzzy time intervals (see [159] for a complete review) and fuzzy locations (e.g. fuzzy polygons) (see [17]). Such a service would make extensive mobile calls like ("Where are you? have the others arrived? where have they met exactly?") obsolete.

Summary

In this chapter we investigated and suggested four applications that have great benefit from using group models. First we suggested several alternatives for indication of groups (visualization and textual indication). It was discussed that group models allow for more efficency in the presentation of social relations with respect to awareness and communication. Second application suggested was an improvement for Collaborative Filtering. It was argued that groups are the natural base sets to use for CF which will have some interesting effects on the quality of CF. The third application we discussed was location based information push. We argued that incorporating group models enlarges the data basis which can be used to create the profiles for filtering, that groups can have a normative effect on information needs and that Ad-Hoc-Groups combined with other group models increase the chance to monitor real social structures and to produce less artifacts. Finally, we presented the idea for a context sensitive group calendar / reminder which also profits from group models.

Chapter 7

Conclusion

7.1 What was Achieved?

The thesis aimed at developing methods for the detection and modeling of Ad-Hoc-Groups and abstract groups in a mobile community scenario. While the discussed algorithms and strategies are not absolutely bound to a mobile community scenario, this framework will ensure that the required data and infrastructure is available and stabilizes the results by providing a social frame of people with common pursuit.

Chapter 1 gave a condensed overview over communities and mobile communities. The achievement of this chapter was to integrate and further develop points of view from numerous scientific sources and to construct a uniform classification scheme and notion system which allows to describe and discuss any type of community from the very abstract level to the level of concrete services.

The contribution of **Chapter 2** was to review what socio-psychology and mathematical psychology provide with respect to defining the notion of a group. By aggregating various publications, a uniform set of characteristics of a group was worked out which represents the smallest common denominator with respect to the definition of group. It was also investigated how mathematical psychology (Sociometry) models and derives group structures and an interesting connection was drawn from crisp hierarchical clustering methods to the graph theoretic group definitions of Wassermann et. al. Furthermore, the chapter successfully attempted to derive criteria on how spatial proximity and self perception influence groups. Thus the basis was laid for conclusions that act as an agenda for the rest of the thesis and which justify the approaches used in chapter 6.

Chapters 3 and 4 were devoted to investigate what types of data were suitable to be subject for the characterization (detection and modeling) of groups and how these data could be acquired, modeled and compared.
The main contribution of **chapter 3** was the thorough development of the SUMI (Simple Urban Mobility Simulator) model. While first providing a general classification of types of available data for user characterization, the main section is devoted to discuss the stochastic models of SUMI in detail. The two main benefits of SUMI are that it delivers very realistic and accurate location and velocity data without the need for acquiring them in field studies and that it delivers precise data about the Ad-Hoc-Group and corresponding abstract group structure at all times of the simulation. This is ideal for quantitatively evaluation group detection and modeling

procedures. The exclusive feature of SUMI is that it simulates group motion and individual motion at the same time while switching between these two basic mobility modes in a seamless and realistic fashion.

Chapter 4 gave several new contributions. First of all, interests were thoroughly investigated for the first time as excellent examples for explicit textual self information. For the first time it was recognized that the phrase structure of free text interests provides an excellent way to reduce the complexity usually found in natural language texts while at the same time being highly expressive. Several test collections were collected (partly by downloading and partly through own survey) to allow for subsequent analysis and algorithm development. The test collections are also used in other scientific projects [176]. The main contribution is the similarity measure for sets of free text interest phrases and for list-of-choice interest vectors. Both measures are a thorough combinations of known resources (such as Encarta or WordNet) and approaches from literature and own developments and ideas. The measures have been quantitatively investigated through comparison with an extensive survey which was thoroughly planned and conducted with the help of over 30 participants. The results were very good, showing that the computational similarity measures matched the human judgment on user-user-similarity with respect to interests very well. The last part of chapter 4 was devoted to the development of a similarity measure for communication data with tree-like structure. What is most remarkable about the result is that it combines standard content analysis techniques with a completely new approach that respects the algebraic graph properties (reply relation and tree structure) of the communication data. Other similar similarity relations in the literature only regard one of the two aspects.

In **chapter 5**, the preparations of the preceding chapters were condensed in the development of procedures to detect and model Ad-Hoc-Groups and abstract groups. A main contribution of the **first part** was to point out that social cluster selection and validation strategies need to be developed in order to successfully detect clusters with respect to highly dynamic context parameters such as location & velocity. The performance of the various alternatives for strategy parameters were thoroughly investigated and the overall performance of the Ad-Hoc-Group detection and modeling strategy was tested against the SUMI simulation data and delivered excellent results. A further interesting development of that chapter is the strategy to compute underlying abstract groups on the basis of member structure and periodicity calculations where periodically occurring Ad-Hoc-Groups are assumed to have a higher probability of being instantiations of a common abstract group than non-periodically occurring Ad-Hoc-groups. This algorithm was also tested against the simulation and showed a good performance.
The **second part of chapter 5** was devoted to the discussion of group detection and modeling on the basis of interest and communication similarities. The only algorithm found in the literature that was capable of computing fuzzy clusters on the basis of relational data alone proved to be faulty. Two innovative approaches were introduced that greatly improved the performance of this algorithm. One was based on Simulated Annealing, the other was based on the observation that clustering could also be performed directly by taking the relational data as patterns. Both approaches were very successful on artificial test data.

Chapter 6 presented four new approaches to incorporate group models into mobile community communication and information management. The group indication type of applications is remarkable because it shows that group models can have a benefit "on their own" by selecting an appropriate presentation technique which is context-aware in the optimal case. The Collaborative Filtering (CF) type of applications showed how group models can be used to control information flows and that groups can be important as actors and recipients of information

flows. The third type of application shows that Ad-Hoc-Groups (especially in combination with other group models) can be excellently used to determine local information needs and can thus be well used for context sensitive information push. The last type of application is related to the first and third type and shows a more conventional type of application that can be used for the organization of team activities in a mobile community setting as well as in a commercial scenario.

7.2 Critical Discussion and Open Questions

Besides the positive and innovative aspects there also aspects which may draw critique and aspects which are not treated in the thesis but deserve being investigated by future work.

The most obvious field for critique and the most urgent open field is the question of **privacy and security**. What has to be said, is that privacy and security matters were not a topic in thesis due to space and time restrictions. Nevertheless these are key topics which must be addressed before deploying any of the described methods or applications. The approaches presented here require an open service and data architecture with unrestricted access in principle. How can privacy and security be enforced in such an open scenario? First of all, a user must be given full access to all of its profile data and must be able to deactivate any service using his data at all times. Trust in the community plays a significant role without no such service will ever have a chance of being deployed. In the COSMOS project it turned out [181] that appropriate levels of context-aware privacy declarations are not easy to realize and require substantial effort from the theoretical side as well as from the organizational and user-interface side. The methods and services introduced here are **exclusively** intended to provide an added value for the user and especially for groups of users and may by no means be abused for monitoring and surveillance purposes, unwanted commercial exploitation, spam etc..

A second field of critique is that the influence of **location accuracy** on the proposed and investigated methods is not further elaborated on in this thesis. COSMOS showed that even the most innovative yet simple and useful service won't function if the accuracy of the used localization technique is not sufficient. In the thesis we assumed, in principle, infinite accuracy for e.g. the location data. Even with a Galileo module in every mobile device it will not be possible to achieve infinite accuracy. We are confident, that the next level of mobile devices will be equipped with location technology that is accurate enough to allow for all proposed methods to function properly. However, it remains an interesting open question what consequences a decrease in accuracy will have for the proposed strategies. Furthermore, no attention has been paid to a (likely) scenario where mobile devices with various different localization accuracies are used in the mobile community at the same time. This is a very interesting field of study that needs to be addressed in the near future.

Furthermore attention has to be paid to the **computational cost** of the proposed group detection and modeling approaches which was not discussed in the thesis. For example, the optimal solution to the clustering problem can be shown to be NP-complete (see [37]) and the semantic similarity measures also require substantial computing time. What does that mean? First of all, it needs to be emphasized that neither of the proposed algorithms requires exponential time. They are all polynomially complex with respect to computing time. That means although we might not achieve an optimal solution, we can be as good as required for the ideas of the thesis. Nevertheless research on incremental methods that update their models when new data occurs

instead of computing the clusters from scratch is a very interesting option. Secondly for most of the longer running computations such as those involving semantic resources like WordNet intermediate results can be stored and reused. Furthermore these approaches involve abstract groups which are not that dynamic and can be computed when the server load is low. Thirdly, all the considerations should be understood in the framework of a mobile community with enough members to make sense but that is restricted in extension and intension. Most of the methods won't function well if used on the scale of the web or by a large telecommunications company because the organizational and intentional brace that a community represents for its members would be missing. It is this frame that gives the proposed approaches additional strength and value.

Another critique aims at the question of **general feasibility and information culture**. "Do we really need that?" "is the information society already developed enough for such services to have a circle of users or even a commercial perspective?" are questions that can be asked when being presented the ideas of this thesis. These are open questions and need to be discussed. Many services such as SMS have become really important (socially as well as economically) although nobody would have expected this. One can assume that SMS would not have been so successful 15 years ago since the culture of information access and information communication has substantially changed in the last decade.

Another open point and subject to critique is why the **group detection methods with respect to interests performed in a rather obscure way**, although the results from the similarity measures were so encouraging. The most obvious candidates are the relational fuzzy clustering approaches used. Perhaps it could be interesting to investigate the introduction of a metric space for the data that would allow the use of conventional fuzzy clusters which are better understood. This point needs further investigation.

7.2.1 Future Prospects

The discussed topics in this thesis imply several possible fields of study for the future. Several disciplines can deliver contributions to the question how or whether at all (Ad-hoc-)groups should become first class entities in the field of supporting social processes especially in supporting mobile communities.

Social psychology could investigate the phenomenon of mixed real word virtual mobile communities in more depth and could especially work out empirically, what types of Ad-Hoc-Groups arise in connection with mobile communication and what the spectrum of their structure, purposes and focus is like. Especially the role of location as a common denominator should be investigated in more depth.

Applied Computer Science can deliver contributions by building prototype systems which make use of (Ad-Hoc-)group detection and modeling in form of mobile services that can be evaluated in field tests. In an iterative process these tests allow for a better adaptation of the services to the emerging consumer needs. This process can generate commercially usable additional demands and added value of mobile communication and mobile computing for the customers with respect to group level support services.

Geo-Information Science could investigate means to integrate different means of location technology and geo-spatial data sources and access services into an easy to use standard. This standard would allow for a uniform access to geo-spatial services such as geo-coding and reverse-geo-coding and would also allow for a standard way to represent and handle locations in var-

ious degrees of accuracy. Such a standard would be a helpful contribution in terms of a more widespread use of location based services in general and especially for location based (Ad-Hoc-)group detection and modeling.

Natural Language Processing techniques could be improved with respect to semantically comparing natural language text elements with a phrase structure such as the free text interest phrases. It is of great importance in view of unobtrusive context sensitive mobile services to automatically extract as much information as possible about a user's interests as possible. A semantically rich similarity measure is also of special significance for group detection which has been shown in the course of the thesis.

Posed questions in the field **Artificial Intelligence and theoretical Computer Science** include the stability of fuzzy and crisp clustering algorithms, the construction of suitable incremental clustering approaches and the further development of relational fuzzy clustering algorithms. In terms of the construction of incremental clustering approaches it would be of special interest for services that have to perform clustering on data sets that evolve over time to have a clustering algorithm that could use the results of clustering run at time t_i in calculating the results at time t_{i+1} in an efficient way. While this could certainly be achieved as a first try by using the results (U, Π) of run t_i as a starting point in one of the discussed iterative algorithm, no systematic results exist about the stability of the resulting clustering. While one would expect the cluster results to change continuously if the data only slightly changes from t_i to t_{i+1} this is not always the case as some of our experiments [37] show.

The above suggestions show that the thesis has brought up interesting questions which are worth investigating in future research.

Appendix A

Ontologies

In order to facilitate the exchange of operational information (knowledge), an agreement on a formal framework for the representation of knowledge, especially of the vocabulary of a knowledge domain, has to be found. This formal framework can be called an **ontology**. In the context of Semantic Web [174, 29] ontologies have become an important field in computer science. For a more thorough introduction into the field, the reader is referred to all articles cited in this section, and especially to the excellent book [29].
There are various definitions for ontology:

- Ontology as a philosophical discipline which deals with the nature of being or the kinds of the existing [174, 43]. This discipline goes back to the age of Aristoteles ("Metaphysics" IV, 1) [174].
- "An ontology defines the basic terms and relations comprising the vocabulary of a topic area, as well as the rules for combining terms and relations to define extensions of the vocabulary." [146](in [48]).
- "An ontology is a hierarchically structured set of terms for describing a domain that can be used as a skeletal foundation for a knowledge base"[180] (in [48]).
- "An ontology is a formal explicit specification of a shared conceptualization" [179] (in [48, 174])(going back on a similar definition by Gruber, 1995).

The heterogeneous nature of these definitions from different sources shows that there is little agreement on a formal definition of ontology (even in a scientific community that aims at supporting the exchange of knowledge through the construction and examination of explicit terminological frameworks (which in turn shows the difficulties connected with that task)).
The first definition is rather a classification of the term with the point of view of a scientific field different from computer science.
The second definition stresses that an ontology defines a **vocabulary** which distinguishes it from knowledge representation formalisms with differing goals. Nevertheless it also defines it as also consisting of **relations** and **rules** which puts an emphasis on the fact that an ontology is generally more than a hierarchy of terms that designate subjects or topics and emphasizes its **operational** character ("[...] combining [...] to define extensions [...]").
The third definition or opinion emphasizes that the set of terms defined in an ontology is **hierarchically structured** which stresses "is-a" relations upon the terms. It is also **domain**

oriented which is an important aspect of an ontology because it is highly modular and dependent on the domain it is intended to be used for. Therefore, there is not one ontology but rather many (subjective) ontologies. Furthermore, the definition stresses that an ontology is an integral part or prerequisite for a knowledge base but that a knowledge base is a more general concept. This aims at separating the topic hierarchy from the logical framework that makes it operational. A definition which has a rather similar bias is due to Gruber (cited in [43]): "Formally, an ontology consists of terms, their definitions and axioms relating them".
Definition 4 is the most agreed upon definition in computer science. It states that an ontology is **formal** and **explicit** which ultimately means machine processable (from a computer science point of view), and that it is a specification of a **shared** conceptualization, which means that the ultimate goal of an ontology is to create a basis on which an exchange of knowledge can take place via a **common** formalism. (Ontologies define "common languages"). A **conceptualization** consists of a set of objects, concepts and other entities about which knowledge is being expressed and of relationships that hold among them" [43]. (Ontologies give structured representations of knowledge). There are more (often rather spongy) definitions of conceptualization in the literature. The degree of formality of the specification of an ontology greatly varies (according to the field of science that it is used in):

- "highly informal: expressed loosely in natural language.
- semi informal: expressed in a restricted and structured form of natural language, greatly increasing clarity by reducing ambiguity [...]
- semi-formal: expressed in an artificial formally defined language. [...]
- rigorously formal: meticulously defined terms with formal semantics, theorems and proofs of such properties as soundness and completeness [...] "[186]

Besides the degree of formality there are a lot of other classification dimensions for ontologies [43], such as

- the degree of generality (e.g. whether the ontology is a generic, common sense ontology or a domain specific one)
- the ontology's design process (e.g. bottom-up or top-down)
- the structure of the taxonomy of terms (e.g. is there a taxonomy at all, is it a tree-like **is-a** taxonomy or a set of such trees etc.)
- structure and representation of the concepts and relations
- structure and representation of axioms and inference-mechanisms (e.g. type of logic used (first order predicate logic, F-logic [82], description-logic [191, 42] etc.), its power of expression, computational limitations etc.)
- intended field of application (e.g. natural language processing, information retrieval, enterprise modeling etc.)

and more. There are countless difficulties that arise along all these classification dimensions. A simple conceptualization of a domain, for example, could simply consist of a is-a hierarchy

of concepts that are relevant for that domain. A concept is understood as a unary predicate $c : \mathcal{U} \to \{0, 1\}$ over a universe of discourse $\mathcal{U}$. In order to define it, we usually need necessary and sufficient conditions, which means that we need a set of super-concepts $\{c_1^u, c_2^u, \dots\}$ (which correspond to the sufficient conditions) and a set of sub-concepts $\{c_1^l, c_2^l, \dots\}$ (which correspond to the necessary conditions) such that $(c \to c_1^u \wedge c_2^u \wedge \dots) \wedge (c_1^l \wedge c_2^l \wedge \dots \to c)$ is true under that interpretation over $\mathcal{U}$ that assigns the specified sets to the symbols c, c_i^u, c_i^l. That would partition $\mathcal{U}$ into a hierarchy of concepts (extensionally defined by sets) where the (binary) relation is-a would implicitly correspond to $\subset$. Unfortunately, not every relation between terms $x \in \mathcal{U}$ is an is-a relation (E.g. (using an object oriented language) besides abstraction (**is-a**) we also have association (**linked-to**) and aggregation (**part-of**)). Therefore a next step in expressiveness would have to include relations (predicates) of any arity (especially > 1). (See [191] for an analysis of workarounds on the basis of description-logic). Now the question arises how statements about relations can be incorporated into the formalism. By allowing relations as first-order objects (reification) one leaves the first order predicate logic framework with all the computational consequences and difficulties. Instead of covering all such aspects and aspects like generation and construction methodologies for ontologies, possible fields of application etc. in depth, which would be impossible within this thesis, we will take a brief look at some examples and formalisms for ontologies in order to give a basic understanding of the subject.

A.1 Examples

There are many existing ontologies which have been described in literature (see [48],[174],[43] for more). Relations which are used most are abstraction relations and part-whole-relations (e.g. **element-of**, **exhaustive partition**, **part of** etc.)[48].

CYC (the name is derived from enCYClopedia), going back to an initiative of people working in the field of Machine Learning (like Lenat), is a comparatively large generic common-sense ontology developed mostly for semantic information retrieval and natural language processing in general. It differs from WordNet in that there is not one tree-like hierarchy of terms but rather basic classification directions along which elements are characterized. In an object oriented terminology one could say that this has the flavor of multiple inheritance. The logic formalism of CYC goes beyond first order predicate logic (e.g. it allows for statements about relations (reification))

TOVE (Toronto Virtual Enterprise) is an ontology developed to model enterprises. It relies on first-order-predicate logic for the formal representation of its logic structures. It consists of several sub-ontologies which model different aspects of enterprises.

KIF (Knowledge Interchange Format) is a formalism which (besides the interchange of more general theories) can also be used for the specification of ontologies. Due to the fact that it provides micro-theories (e.g. for numbers and sets) it can be considered a general ontology itself. Other systems such as **ONTOLINGUA** (which aims at providing a toolset for unified views on ontologies and handling ontologies) rely on KIF as a representation formalism.

A.2 Ontologies and the Semantic Web

Based on the XML family of technologies which aim at separating content, structure and layout, the Semantic Web is a vision to enhance Web-Technology with frameworks that allow for semantics to be specified which are concretized and bundled in form of standard-recommendations by the W3C. An excellent introduction is provided by [10] and [29].
The Semantic Web's layers on top of XML aim at providing the infrastructure for logic based semantic reasoning about resources on the Web which may allow for intelligent information retrieval agents on the web, enhanced knowledge management (sharing, reuse, etc) in large companies and many more visionary applications.
The basis of the Semantic Web is RDF [66] an assertional language / data-model which basically allows to make statements about resources and express binary predicates (relations between resources). Its companion recommendation RDFS [66] is a schema language using RDF which represents a basic vocabulary for the description of properties and classes of RDF Resources and a semantics for is-a hierarchies among such properties and classes [116]. Together they allow for the build up of simple ontologies in the fashion of early AI's semantic nets and frames. For further expressiveness the Web Ontology Language OWL [116] was created. On the sound theoretical basis of Description Logics [42] and previous experiences with the languages DAML and OIL (see [116]), the language provides 3 levels of expressiveness (OWL Lite, OWL DL, OWL Full) with increasing expressive power and increasing reasoning complexity. OWL adds structures which allow to express more relations between classes such as disjointness, cardinalities, equality, characteristics of properties etc. [116].
A large number of interesting tools for the engineering of ontologies has been built which include ontology editors, reasoning engines and many more (See [67] for typical Use-Cases).

Appendix B

Spelling Correction

Spelling errors in texts can be corrected by means of sophisticated spelling correction techniques (see [94] for an excellent review). The most basic task is so-called **non-word-error detection** [94]. A non-word error is an error which leads to a string which is not a word (e.g. "Bisketball"). These types of errors can be detected by either **N-Gram**-Analysis, which efficiently compares all N-ary substrings (all N-Grams) of a word against a lexicon of allowed N-Grams, or compares all words efficiently (e.g. by hashing) against a lexicon of allowed words. These **morphological lexica** contain a substantial part of the words of a language plus all forms of this word that result from **morphological processes** [112] like **inflection** (systematically modifying a root form of a word (e.g. by means of prefixes or suffixes etc.) in order to express a different tense, number, casus etc.).
The next task which is more difficult is to **automatically correct** the errors found. One can distinguish between **isolated word correction** and **context sensitive correction** techniques[94]. There are many techniques for isolated word error correction [94], where minimal edit distance is one of the simplest approaches, yet performing considerably well[94]. Edit distance correction will correct any non-word to the word in the morphological lexicon which has the smallest edit distance. Edit (Levenshtein) distance $d_L(w_i, w_j)$ between words $w_i = (c_{i_1}, c_{i_2}, \ldots, c_{i_m})$ and $w_j = (c_{j_1}, c_{j_2}, \ldots, c_{j_n})$ is defined as [161]:

$$d_L(w_i, w_j) = d_L((c_{i_1}, c_{i_2}, \ldots, c_{i_m}), (c_{j_1}, c_{j_2}, \ldots, c_{j_n})) =$$
$$\begin{cases} m & \text{if } n = 0 \\ n & \text{if } m = 0 \\ \min\{d_L((c_{i_1}, c_{i_2}, \ldots, c_{i_{m-1}}), (c_{j_1}, c_{j_2}, \ldots, c_{j_n})) + 1, & \\ \quad d_L((c_{i_1}, c_{i_2}, \ldots, c_{i_m}), (c_{j_1}, c_{j_2}, \ldots, c_{j_{n-1}})) + 1, & \text{else} \\ \quad d_L((c_{i_1}, c_{i_2}, \ldots, c_{i_{m-1}}), (c_{j_1}, c_{j_2}, \ldots, c_{j_{n-1}})) + z(c_{i_m}, c_{j_n}), \} & \end{cases} \tag{B.1}$$

and measures number of deletions, insertions, or substitutions required to transform w_i into w_j (we only regard lower case charcters $c \in \{$"a","b",...,"z"$\}$). This definition is equivalent to a dynamic programming algorithm for the computation of the Levenshtein distance with average and worst time complexity $O(|w_i||w_j|)$ and space complexity $O(\min(|w_i|, |w_j|))$ where $|w|$ denotes the length of the word [144]. Numerous alternative algorithms with improved complexity have been developed [144]. E.g. the Levenshtein distance between the words "dancing" and "dance" is equal to 3 which is less than the Levenshtein distance between e.g. "dancing" and "shoe" which is 7. The maximum Levenshtein distance between words w_1 and w_2 is $\max(|w_1|, |w_2|)$. In

that definition the function $z(c_x, c_y)$ measures the similarity between characters c_x and c_y and is usually defined as $z(c_x, c_y) = 1 - \delta_{c_x c_y}$. Other functions that include phonetic similarities can be used as well [161].
Besides non-word spelling errors there are errors that lead to words which are correct word forms but have a different semantics (**Real-Word-Errors**)[94]. These types of errors can occur by chance or via "psycholinguistic" errors like incorrectly using the spelling of a Homophone or near-Homophone word "peace vs. piece". They are much harder to detect and even harder to correct. Detection and correction require an analysis of the word's **context** which means that the words surrounding the word in question needs to be analyzed syntactically, grammatically and semantically (See [94] for a brief introduction).

Appendix C

Interest Test Collections and Similarity Survey

In chapter 4, similarity measures for interest phrases were discussed. It was described, how some collections of interest phrases were collected. For the sake of completeness, the 2 collections "Survey Collection" and "Dating Collection" which were investigated in detail are presented here in figures C.1, C.2, C.3 and C.4.

Additionally, in order to gather data about how people judge similarities between persons on the basis of sets of free text interest phrases, list-of-choice interest vectors and single free text interest phrases, 3 surveys were conducted.
30 test-persons from the personal social environment of the thesis author were emailed a questionnaire. 10 persons were mailed the questionnaire for judging similarities between persons on the basis of sets of free text interest phrases. 10 persons were mailed the questionnaire for judging similarities between persons on the basis of list-of-choice interest vectors and 10 further persons were mailed the questionnaire for judging similarities between persons on the basis of single free text interest phrases.
For a discussion of the results see chapter 4.
Figures C.6, C.7 and C.8 show the first parts of the questionnaires for the three surveys.
Figure C.5 shows the list of single free text interest phrases used for survey 3.
Table C.5 shows, which elements of the "Survey Collection" and "Dating Collection" and the list of free text interest phrases (figure C.5) were to be compared in the respective surveys.

	Survey 1	Survey 2	Survey 3
Numbering according to Figure	C.1	C.3	C.5
Elements $\xleftrightarrow{\text{compared with}}$ elements	11 - 15 $\leftrightarrow$ 1 - 5 16 - 20 $\leftrightarrow$ 1 - 5 11 - 15 $\leftrightarrow$ 6 - 10 16 - 20 $\leftrightarrow$ 6 - 10	11 - 15 $\leftrightarrow$ 1 - 5 16 - 20 $\leftrightarrow$ 1 - 5 11 - 15 $\leftrightarrow$ 6 - 10 16 - 20 $\leftrightarrow$ 6 - 10	21 - 30 $\leftrightarrow$ 1 - 10 31 - 40 $\leftrightarrow$ 1 - 10 21 - 30 $\leftrightarrow$ 11 - 20 31 - 40 $\leftrightarrow$ 11 - 20

Table C.1: The elements from the two collections "Survey Collection" and "Dating Collection" and the list of free text interest phrases (figure C.5) used in the surveys 1, 2 and 3

The "Survey Collection" (Part I)

(1) reading; windsurfing; computers; lying at the sea; biking

(2) footbal; newspaper; music; mobile phones; sports; tv; movies; computers; women; drinking beer

(3) skiing; dancing; sailing; cycling; cinema; foreign languages; reading; programming

(4) volleyball; soccer; jogging; inline-skating; playing drums; drawing; meeting friends; visiting exibitions; reading books; spead reading; party; playing cards and board games; icq'ing; playing tennis

(5) sailing; books; tabletennis; juggling; friends; games

(6) skiing; listening to bach and beethoven; esp piano music; travel; food

(7) music; reading; books; cars; motorracing; swimming; hiking; cycling; hanging around; food; architecture; traveling; sleeping

(8) jogging; music; sports; mountains; traveling; computer; books; movies; sub-culture; theater; writing texts; friends; parties; festivals; art; painting

(9) cooking; kung fu; jogging; reading; watching tv; hiking; biking; swimming; singing; trash metal; cars; meeting friends; modern arts

(10) money; sex; wisdom; traveling; sea; sports; reading; computer; talking; drinking; movies; humor; childs

(11) computer; photo; model railway; football; music

(12) reading; books in general; traveling; especially oriental countries; religion; christianity; getting to know human beings; talking with friends and my family; cooking; good wine; roleplaying like DSA and D´n´D; YMCA; honarary works; camping; rollerblading; photography; dancing; singing; playing guitar; moving; sleeping; daydreaming

(13) soccer; canyoning; beach volleyball; cycling; programming; java; frag georg; j2ee; michel friedman

(14) jazz improvisation; modern music; baroque music; lied; operating systems; linux; digital audio processing; information systems engineering; systems administration; networking; network security; free software; internet communities; contemporary english literature; poetry; theatre; modern art

(15) reading; chess; hiking; music; poetry; meditation

(16) soccer; badminton; boxing; music; good food; travelling; clubbing; going out for a drink; dancing; hanging loose; discussing; dreaming; hiking; going to the movies; thinking; reading

(17) baseball; sport; skiing; music; travelling; soccer; photography; tolkien

(18) reading books; going to the movies; playing golf; listening to music; being with monika; ballroom dancing

(19) music; zeitgeist; love; friends

(20) meeting friends; cinema; jogging; aerobic; listen music; dancing; reading; watching tv; star trek; visiting my family; musicals; cats

(21) skiing; sailing; surfing; swimming; musik; (electrified) string-instruments; biking; hiking; working; project management; coaching; value webs; macroeconomic policy; interest rates; information systems; computer science

(22) playing the saxophon; sports; drinking beer; bicycle trips; jazz-music; gis; geography; travelling

(23) dating with friends; reading novels; skiing; playing board games; jogging; beeing online; having sex with my girlfriend; trying to find the question for the answer 42; lazing about; corresponding worldwide with penfriends; watching TV; partying; going abroad; having wanderlust

(24) reading; music; skiing; traveling

(25) sport; mountains; climbing; running; party; nature; adventure; skiing; volleyball; reading; books; history; countries; travelling; languages; studying; talking; thinking; sun; wine; beer; girls; music; guitar; singing; acrobatic; soccer; friends

(26) sports (badminton; squash ; tennis; volleyball) ; meeting lots of people ; solving problems (especially mathematical ones) ; music ; having fun ; nice shoes ; being informed ; friends

(27) reading; photography; driving mountainbike; fitness training

(28) food; sleeping; music; tea; concerts; swimming; barbecue

(29) sport; music; party; sleeping

(30) music; sozial science; to drink coffee; be on the beach; have some cocktails

(31) books; reading; sports; jogging; soccer/football; drinking; the internet

(32) reading; snowboarding; meet my friends; go out; go to the movies; Spanish Literature; Mexico; dogs; snorkeling; internet auctions

(33) meeting friends; music and dancing; jogging; biking; swimming; reading; playing clarinet

(34) going to the theatre (cinema); swimming; reading; hiking; hanging around; listening to music; meeting friends;

(35) kissing beautifull girls; reading exiting books; often i am producing mp3; sometimes i go waterskiing; indoor climbing; writing computerprograms; driving cars; riding fast motorbikes; mountainbike downhill; playing computergames; diving in the read sea

(36) party; good food; jogging; downhill; backcountry skiing and snowboarding; sailing; golf; traveling; cooking; movies; music; international politics; hiking; snorkeling

(37) riding; reading; biking; sleeping; playing piano; gathering picturebooks

(38) traveling, sailing, diving, trekking, driving motorbike, swimming, computer, internet, games

(39) computer; cinema; pen & paper rpgs; computer games; listening to music; babylon 5; terry pratchett

(40) read; write poems and literature; travel; languages; computer; science; soccer; partying; cars; long and interesting talks; getting to knwo people; hiking; biking; sailing; music;

(41) riding; mountain biking; motor biking; taking the dog out; mountaineering

(42) reading; sleeping; movies; politics; newspapers; eating; talking; classical music; history; modern french and american literature; languages; travel

(43) reading history; reading novels; baking; cooking vegetarian meals; local grammars; history of science; history of physics; Russian history; history of Russian science; travel; bad television dubbed in foreign languages; Simpsons trivia; Hungarian language; empire; choral singing; classical music; speculating about why I am so good at wasting time; current events; left-wing blogs; keeping an eye out for Hugo Boss clothing on sale; Hans Eichel's latest addition to his collection of piggy banks; learning html without actually working at it; writing obscure books no one will read; writing a bestseller; watching silly movies; wondering why so many of my interests are listed in the form of gerund phrases

(44) roleplaying; reading; cycling; movies; ancient history

(45) waterpolo; sport; politics; lenguages; cycling

(46) computing; science; biology; mountain-biking; object-oriented software development; java; cooking; travelling; sports; reading; motorbiking; cinema; videos; sexuality; car-driving; skiing

(47) wine; thinking; contemplation; visiting other countries; working in the garden; foundations; love; sun; go swimming; eating chocolat; talking with friends; singing in the rain; working with my hands

(48) mathematics; dancing; european movies; cooking; foreign langauges

(49) writing poems; playing drums; snowboarding; meeting my girlfriend; watching films at the cinema / at home; reading philosophical / historical / ... books; bathing; dancing

(50) spending "quality time" with my son (=family); playing the violin with my son; singing in the church choir; reading english historical novels; reading english fiction; reading anything interesting and humorous; earning money; so that my family may not starve; riding my bicycle; repairing my bicycle; skiing in winter; going to the opera

Figure C.1: The first 50 free text interest phrases sets from the "Survey Collection"

The "Survey Collection" (Part II)

(51) shopping in the city; watching movies; relaxing at home; in the nature or at nice parties; reading interesting literature; painting on canvas; surfing in the internet; listening to good music (modern; classic); eating at good restaurants; discussing with good friends; apline skiing in the winter; playing tennis in the summer; going to the fitness-studio and taking part in the "fat-burn-program"; visiting conferences

(52) travel; reading; watch TV; movie; eating chinese food; books and internet; cafe(house); swimming

(53) reading; swimming; walking

(54) reading books; listening music; going out with friends; discuss life; love; wishes etc.; traveling in really different culturs; playing card game; volleyball; watching TV sunday evening; theatre; programm cinema

(55) common affairs; reading books; hang out with friends

(56) skiing; swimming; reading; watching films; travelling to interesting countries; eating good meals; meeting with my friends

(57) trekking; traveling (visiting all countries all over the world); doing sports (table tennis; soccer; volleyball); watching tv sport (winter sports; soccer; handball;....); making Music (singing; piping); playing computer games (counterstrike;...)

(58) taking photos; playing volleyball; playing piano; hiking; biking; listening to music; going clubbing

(59) sex; reading books; reading "Der Spiegel"; politics; historie; sex

(60) chess;cinema;biking;swimming;skiing

(61) business; technology; basketball; movies; hamburgers

(62) singing; walking; nature

(63) flying small aircraft; jogging; travelling; mathematics; economics; music; biking; inline skating; skiing; badminton

(64) reading; photography; meeting friends; cinema

(65) squash; football; movies; going out with friends; scuba diving

(66) playing squash; badminton; chess; piano; dancing tango; have parties

(67) travel; books; diving; sailing; sleeping

(68) listen to music; programming web sites; having fun with my girlfriend :-)

(69) reading; listening to music; basketball; snowboarding; sports in general; beeing with people

(70) girl friend; swimming in the swimclub; reading acticels about moral and etic; going to clubs; enventing business ideas; making money; survival in the daily work inviroment

(71) playing computer games; jogging; cooking; travelling; getting laid; making music; going to clubs

(72) wendy

(73) informatics; going to the movies with friends; listening to music; reading books; success in programming lange systems; cooking and eating good stuff

(74) travelling; playing boardgames; swimming; basketball; party

(75) reading newspapers; travelling around the world; listening to good music; talking to friends

(76) reading interesting books; walking in the rain; singing the hole day; biking; thinking about the live; live musik

(77) go windsurfing; playing chess; painting; modeling human being; visit foreign countries; playing with my daughter / dog

(78) downhill mountainbiking; internetsurfing; dirtjumping; highjacking

(79) playing guitar; singing with a warm voice; biking in canada; taking photos of sunrises

(80) internet; reading; party; jogging

(81) watching the game; trinking a bud; internet surfing; writing and reading email; playing computer games; driving car; having fun with my girlfriend; watching movies; reading books; study

(82) playing egoshooter games online; watching "happytreefriends" episodes; watching movies on tv

(83) sports; tv; games

(84) reading books; playing per pc; biking; talking about various things; beeing with friends; painting; writing; listening to music

(85) playing football; reading books; meeting friends; playing squash; playing tennis; going for a drink

(86) listening to music; going to arts exhibitions; visiting friends; going to cafés; reading books in foreign languages; snowboarding; travelling

(87) wing tsun kung fu; playing guitar; listening to suzanne vega; eating; cooking chinese food

(88) reading books; watching movies; sports (basketball; running); listening to music; working

(89) going out with friends; waching the sun going down while having a drink; loosing control; talking to wendy on the phone

(90) swimming; travelling the world; reading books; skiing; snowboarding; hking; visiting friends

(91) philosophy; computer; jogging; skiing; mountainclimbing; music; books; films; parties; people; friends; woman

(92) reading surprising and exciting books; eating well tasting meals; watching serious films; sometimes drinking a glass of wine or tequila; singing(mostly and louder while i am alone at home); meating friends in a club or a disco; swimming in summer in a lake or the see; list could be much longer but i think it is enough!

(93) playing realtime strategy games (online); cooking; playing chess; designing web pages

(94) chatting; reading English books; watching tv; talking with friends; walking in the countryside

(95) playing computer; watching TV; eating; sleeping

(96) movies; playing billard and chess; geocaching; robotics; working for a web-community; reading

(97) chatting and surfing on the internet; going to the movie theatre; making music and listening to music; reading science-fiction books; having anal sex

(98) reading interesting books; singing in the rain; painting; biking in the alps

(99) reading science and computer books; contributing to open source projects; working with my computer; making music; hanging around with friends

(100) reading; snowboarding; going out in the evening; cooking; jogging; fitness training

Figure C.2: Free text interest phrases sets 51-100 from the "Survey Collection"

The "Dating Collection" (Part I)

(1) Dancing, Family, Movies, Listening to Music, Reading, Watching Sports, Theater, Travel, Cooking, Health/Fitness

(2) Arts, Dancing, Dining, Family, Movies, Outdoor Activities, Photography, Watching Sports, Theater, Travel, Cooking, Computers / Internet, Television, Gardening, Playing Music, Playing Sports, Health/Fitness

(3) Dining, Family, Movies, Listening to Music, Reading, Theater, Travel

(4) Arts, Movies, Listening to Music, Reading, Theater, Travel, Cooking, Computers / Internet, Health/Fitness

(5) Arts, Dancing, Dining, Movies, Listening to Music, Outdoor Activities, Photography, Reading, Theater, Travel, Cooking, Computers / Internet, Television, Crafts, Health/Fitness

(6) Arts, Dancing, Listening to Music, Outdoor Activities, Photography, Reading, Travel

(7) Dancing, Dining, Family, Listening to Music, Outdoor Activities, Photography, Reading, Religion / Spirituality, Theater, Travel, Cooking, Crafts, Playing Music, Health/Fitness

(8) Family, Movies, Listening to Music, Reading, Computers / Internet

(9) Arts, Community Service, Dining, Family, Movies, Listening to Music, Photography, Reading, Theater, Travel

(10) Arts, Dining, Family, Movies, Outdoor Activities, Reading, Travel

(11) Arts, Community Service, Dancing, Dining, Family, Listening to Music, Reading, Religion / Spirituality, Theater, Travel, Cooking, Gardening, Health/Fitness

(12) Arts, Dining, Movies, Listening to Music, Outdoor Activities, Reading, Travel, Cooking

(13) Arts, Family, Movies, Listening to Music, Outdoor Activities, Travel, Cooking, Computers / Internet

(14) Dancing, Dining, Movies, Listening to Music, Reading, Watching Sports, Travel, Television, Health/Fitness

(15) Dancing, Dining, Outdoor Activities, Crafts

(16) Dancing, Dining, Movies, Listening to Music, Outdoor Activities, Theater, Travel, Television

(17) Arts, Family, Movies, Listening to Music, Outdoor Activities, Photography, Reading, Theater, Travel

(18) Dancing, Dining, Movies, Reading, Cooking

(19) Movies, Listening to Music, Reading, Theater, Travel, Computers / Internet, Television

(20) Dancing, Movies, Listening to Music, Travel, Playing Music

(21) Arts, Community Service, Dancing, Dining, Family, Movies, Listening to Music, Photography, Reading, Religion / Spirituality, Theater, Travel, Cooking, Television, Health/Fitness

(22) Listening to Music, Travel, Cooking, Computers / Internet, Television

(23) Arts, Dancing, Dining, Movies, Listening to Music, Reading, Theater, Travel, Cooking, Health/Fitness

(24) Dining, Listening to Music, Outdoor Activities, Reading, Travel, Cooking, Health/Fitness

(25) Family, Movies, Listening to Music, Reading, Television

(26) Family, Listening to Music, Reading, Theater, Travel

(27) Dancing, Dining, Movies, Travel, Cooking

(28) Dancing, Travel, Health/Fitness

(29) Arts, Community Service, Dancing, Family, Movies, Listening to Music, Outdoor Activities, Reading, Watching Sports, Travel, Cooking, Computers / Internet, Crafts, Health/Fitness

(30) Dancing, Movies, Listening to Music, Outdoor Activities, Watching Sports, Health/Fitness

(31) Dancing, Dining, Movies, Listening to Music, Outdoor Activities, Reading, Religion / Spirituality, Watching Sports, Theater, Travel, Cooking, Computers / Internet, Television, Gardening, Playing Music, Health/Fitness

(32) Arts, Dancing, Dining, Movies, Outdoor Activities, Reading, Cooking

(33) Arts, Dancing, Dining, Movies, Listening to Music, Outdoor Activities, Theater, Travel, Playing Music

(34) Dancing, Dining, Movies, Listening to Music, Travel, Cooking, Computers / Internet, Health/Fitness

(35) Computers / Internet

(36) Arts, Dancing, Dining, Movies, Outdoor Activities, Reading, Religion / Spirituality, Theater, Travel, Cooking, Playing Music, Playing Sports, Health/Fitness

(37) Dancing, Cooking

(38) Arts, Dancing, Dining, Family, Movies, Listening to Music, Outdoor Activities, Photography, Reading, Religion / Spirituality, Watching Sports, Theater, Travel, Cooking, Computers / Internet, Gaming, Television, Gardening, Playing Music, Playing Sports

(39) Community Service, Family, Movies, Listening to Music, Outdoor Activities, Reading, Religion / Spirituality, Theater, Travel, Cooking, Computers / Internet, Television

(40) Arts, Dancing, Dining, Family, Movies, Listening to Music

(41) Community Service, Religion / Spirituality, Theater

(42) Movies, Travel

(43) Arts, Dancing, Dining, Family, Movies, Listening to Music, Reading, Religion / Spirituality, Theater, Travel, Cooking, Health/Fitness

(44) Arts, Dancing, Dining, Family, Movies, Listening to Music, Outdoor Activities, Reading, Travel, Cooking, Health/Fitness

(45) Arts, Community Service, Dancing, Dining, Family, Movies, Theater, Travel

(46) Arts, Dancing, Dining, Movies, Photography, Reading, Travel, Television, Health/Fitness

(47) Community Service, Dining, Family, Movies, Outdoor Activities, Theater, Travel

(48) Arts, Dancing, Dining, Family, Movies, Listening to Music, Outdoor Activities, Reading, Theater, Travel, Cooking, Gardening, Crafts

(49) Arts, Dancing, Dining, Movies, Listening to Music, Outdoor Activities, Reading, Travel, Computers / Internet, Television, Playing Music, Health/Fitness

(50) Listening to Music

Figure C.3: The first 50 List-of-Choice interest vectors from the "Dating Collection"

The "Dating Collection" (Part II)

(51) Dancing, Dining, Family, Travel

(52) Dancing, Dining, Family, Movies, Listening to Music, Reading, Theater, Travel, Cooking, Computers / Internet, Gaming

(53) Dancing, Dining, Family, Outdoor Activities, Reading

(54) Dancing, Listening to Music, Travel, Cooking, Computers / Internet

(55) Listening to Music, Reading, Computers / Internet, Television

(56) Computers / Internet, Gaming, Television

(57) Listening to Music

(58) Arts, Reading, Theater, Travel, Cooking, Crafts

(59) Arts, Dining, Movies, Listening to Music, Photography, Reading, Travel

(60) Arts, Photography, Reading, Health/Fitness

(61) Arts, Dining, Family, Movies, Reading, Theater, Travel, Crafts, Health/Fitness

(62) Arts, Dining, Family, Movies, Listening to Music, Reading, Theater, Travel, Cooking, Computers / Internet, Playing Music

(63) Arts, Dancing, Movies, Listening to Music, Outdoor Activities, Travel, Cooking

(64) Arts, Community Service, Dancing, Dining, Family, Movies, Listening to Music, Outdoor Activities, Photography, Reading, Religion / Spirituality, Theater, Travel, Cooking, Playing Music, Health/Fitness

(65) Dancing, Dining, Family, Listening to Music, Travel, Cooking, Television

(66) Arts, Dancing, Dining, Family, Movies, Listening to Music, Outdoor Activities, Travel, Playing Music, Playing Sports, Health/Fitness

(67) Arts, Movies, Listening to Music, Outdoor Activities, Reading, Watching Sports, Theater, Travel, Computers / Internet, Television, Playing Music, Playing Sports, Health/Fitness

(68) Community Service, Dining, Family, Movies, Listening to Music, Outdoor Activities, Reading, Watching Sports, Theater

(69) Arts, Dancing, Dining, Movies, Listening to Music, Outdoor Activities, Photography, Reading, Travel, Health/Fitness

(70) Listening to Music, Outdoor Activities, Reading, Travel, Playing Music, Playing Sports

(71) Arts, Movies, Listening to Music, Outdoor Activities, Reading, Watching Sports, Computers / Internet, Television, Playing Sports, Health/Fitness

(72) Dining, Listening to Music, Outdoor Activities, Reading, Theater, Travel, Playing Sports

(73) Arts, Movies, Theater

(74) Movies, Travel, Cooking

(75) Dancing, Movies, Photography, Travel, Playing Music, Playing Sports

(76) Dancing, Dining, Movies, Listening to Music, Photography, Watching Sports, Travel, Cooking

(77) Community Service, Dancing, Dining, Movies, Listening to Music, Photography, Reading, Religion / Spirituality, Watching Sports, Travel, Cooking, Computers / Internet, Television, Crafts, Playing Sports, Health/Fitness

(78) Dancing, Dining, Reading, Crafts

(79) Arts, Dining, Family, Movies, Listening to Music, Outdoor Activities, Reading, Religion / Spirituality, Theater, Travel, Cooking, Computers / Internet, Gaming, Television, Playing Music

(80) Listening to Music, Religion / Spirituality, Cooking

(81) Arts, Dancing, Dining, Family, Listening to Music, Outdoor Activities, Reading, Theater, Travel, Cooking, Health/Fitness

(82) Arts, Dancing, Dining, Movies, Listening to Music, Outdoor Activities, Theater, Travel, Television, Playing Music

(83) Dancing, Dining, Family, Listening to Music, Cooking

(84) Arts, Dancing, Movies, Listening to Music, Outdoor Activities, Reading, Religion / Spirituality, Theater, Travel, Health/Fitness

(85) Dancing, Dining, Family, Movies, Listening to Music, Outdoor Activities, Reading, Religion / Spirituality, Watching Sports, Theater

(86) Dancing, Family, Movies, Listening to Music, Outdoor Activities, Reading, Cooking, Computers / Internet

(87) Arts, Dancing, Dining, Movies, Listening to Music, Outdoor Activities, Reading, Theater, Travel, Computers / Internet, Health/Fitness

(88) Arts, Dancing, Dining, Family, Movies, Listening to Music, Outdoor Activities, Theater, Travel, Television, Playing Music

(89) Family, Reading, Travel

(90) Community Service, Family, Movies, Listening to Music, Outdoor Activities, Photography, Reading, Travel

(91) Arts, Outdoor Activities, Photography, Reading, Theater, Computers / Internet, Playing Sports, Health/Fitness

(92) Movies, Listening to Music, Reading, Theater, Travel

(93) Community Service, Dining, Family, Movies, Listening to Music, Outdoor Activities, Reading, Theater, Travel, Cooking, Health/Fitness

(94) Dancing, Dining, Family, Movies, Listening to Music, Watching Sports, Travel

(95) Dancing, Family, Movies, Listening to Music, Outdoor Activities, Reading, Religion / Spirituality, Watching Sports, Cooking, Computers / Internet, Television, Gardening, Crafts, Health/Fitness

(96) Arts, Dining, Family, Movies, Listening to Music, Outdoor Activities, Reading, Travel, Cooking, Computers / Internet, Television, Health/Fitness

(97) Cooking, Gardening, Health/Fitness

(98) Community Service, Dancing, Listening to Music, Reading, Health/Fitness

(99) Arts, Dancing, Movies, Listening to Music, Outdoor Activities, Travel, Cooking

(100) Arts, Dancing, Dining, Family, Movies, Listening to Music, Reading, Religion / Spirituality, Theater, Travel, Computers / Internet, Television, Playing Music

Figure C.4: List-of-Choice interest vectors 51-100 from the "Dating Collection"

The Free Text Interest Phrases Used for Survey 3

(1) reading	(21) inline-skating
(2) windsurfing	(22) playing drums
(3) computers	(23) drawing
(4) lying at the sea	(24) meeting friends
(5) biking	(25) visiting exibitions
(6) footbal	(26) reading books
(7) newspaper	(27) spead reading
(8) music	(28) party
(9) mobile phones	(29) playing cards and board games
(10) sports	(30) icq'ing
(11) tv	(31) cars
(12) movies	(32) motorracing
(13) computers	(33) swimming
(14) women	(34) hiking
(15) drinking beer	(35) hanging around
(16) skiing	(36) food
(17) dancing	(37) architecture
(18) sailing	(38) sleeping
(19) cycling	(39) mountains
(20) cinema	(40) sub-culture

Figure C.5: The free text interest phrases used for survey 3 which were taken from the "Survey Collection"

Umfrage 1 (Kurz)

Wie ähnlich sind sich die unten stehenden Interessen von Personen?

(Jede Liste von Interessen wurde von einer Person angegeben).

Geben Sie für jedes Paar von Listen eine Zahl zwischen 0 und 10 an, die die Ähnlichkeit quantifiziert!

- **0** bedeutet: gar keine Ähnlichkeit
- **10** bedeutet: die Listen sind identisch (maximal ähnlich)

Bei der Bewertung der Ähnlichkeit können alle Arten von semantischer Beziehung zwischen den Listen-Elementen berücksichtigt werden:

- Abstraktionsbeziehungen (A ist ein Spezialfall von B) ("is-a") (Bsp: B="reading", A="reading Shakespeare"
- Aggregationsbeziehungen (A ist Teil von B) ("part-of") (Bsp: B="car" A="wheel")
- Assoszіationsbeziehungen (A hat allgemeinen Bezug zu B) (Bsp: A="Pinguin" B="Antarktis")
  etc.

Es sollen jedoch nicht einzelne Elemente der Listen miteinander verglichen werden, sondern eine Gesamt-Ähnlichkeit zwischen zwei Listen angegeben werden. Dies ist notwendigerweise eine **subjektive** Beurteilung, solle jedoch so sorgfältig wie möglich erfolgen.

1.3

	reading; windsurfing; computers; lying at the sea; biking	footbal; newspaper; music; mobile phones; sports; tv; movies; computers; women; drinking beer	skiing; dancing; sailing; cycling; cinema; foreign languages; reading; programming	volleyball; soccer; jogging; inline-skating; playing drums; drawing; meeting friends; visiting exibitions; reading books; speed reading; party; playing cards and board games; icq'ing; playing tennis	sailing; books; tabletennis; juggling; friends; games
computer; photo; model railway; football; music					
reading; books in general; traveling; especially oriental countries; religion; christianity; getting to know human beings; talking with friends and my family; cooking; good wine; roleplaying like DSA and D'n'D; YMCA; honarary works; camping; rollerblading; photography; dancing; singing; playing guitar; moving; sleeping; day-dreaming					
soccer; canyoning; beach volleyball; cycling; programming; java; frag georg; j2ee; michel friedman					
jazz improvisation; modern music; baroque music; lied; operating systems; linux; digital audio processing; information systems engineering; systems administration; networking; network security; free software; internet communities; contemporary english literature; poetry; theatre; modern art					
reading; chess; hiking; music; poetry; meditation					

Figure C.6: The first two pages of the questionnaire of survey 1

Umfrage 2 (Kurz)

Wie ähnlich sind sich die unten stehenden Interessen von Personen?

(Jede Liste von Interessen wurde von einer Person angegeben).

Geben Sie für jedes Paar von Listen eine Zahl zwischen 0 und 10 an, die die Ähnlichkeit quantifiziert!

- **0** bedeutet: gar keine Ähnlichkeit
- **10** bedeutet: die Listen sind identisch (maximal ähnlich)

Bei der Bewertung der Ähnlichkeit können alle Arten von semantischer Beziehung zwischen den Listen-Elementen berücksichtigt werden:

- Abstraktionsbeziehungen (A ist ein Spezialfall von B) ("is-a") (Bsp: B="reading", A="reading Shakespeare"
- Aggregationsbeziehungen (A ist Teil von B) ("part-of") (Bsp: B="car" A="wheel")
- Assoziationsbeziehungen (A hat allgemeinen Bezug zu B) (Bsp: A="Pinguin" B="Antarktis")
  etc.

Es sollen jedoch nicht einzelne Elemente der Listen miteinander verglichen werden, sondern eine Gesamt-Ähnlichkeit zwischen zwei Listen angegeben werden. Dies ist notwendigerweise eine **subjektive** Beurteilung, solle jedoch so sorgfältig wie möglich erfolgen.

1.3

	Dancing, Family, Movies, Listening to Music, Reading, Watching Sports, Theater, Travel, Cooking, Health/Fitness	Arts, Dancing, Dining, Family, Movies, Outdoor Activities, Photography, Watching Sports, Theater, Travel, Cooking, Computers / Internet, Television, Gardening, Playing Music, Playing Sports, Health/Fitness	Dining, Family, Movies, Listening to Music, Reading, Theater, Travel	Arts, Movies, Listening to Music, Reading, Theater, Travel, Cooking, Computers / Internet, Health/Fitness	Arts, Dancing, Dining, Movies, Listening to Music, Outdoor Activities, Photography, Reading, Theater, Travel, Cooking, Computers / Internet, Television, Crafts, Health/Fitness
Arts, Community Service, Dancing, Dining, Family, Listening to Music, Reading, Religion / Spirituality, Theater, Travel, Cooking, Gardening, Health/Fitness					
Arts, Dining, Movies, Listening to Music, Outdoor Activities, Reading, Travel, Cooking					
Arts, Family, Movies, Listening to Music, Outdoor Activities, Travel, Cooking, Computers / Internet					
Dancing, Dining, Movies, Listening to Music, Reading, Watching Sports, Travel, Television, Health/Fitness					
Dancing, Dining, Outdoor Activities, Crafts					

Figure C.7: The first two pages of the questionnaire of survey 2

Umfrage 3 (Kurz)

Wie ähnlich sind sich die unten stehenden Einzeln-Interessen ?

Geben Sie für jedes Paar von Einzel-Interessen eine Zahl zwischen 0 und 10 an, die die Ähnlichkeit bzw. den semantischen Bezug quantifiziert!

- **0** bedeutet: gar keine Ähnlichkeit
- **10** bedeutet: die Listen sind identisch (maximal ähnlich)

Bei der Bewertung der Ähnlichkeit können alle Arten von semantischer Beziehung zwischen den Listen-Elementen berücksichtigt werden:

- Abstraktionsbeziehungen (A ist ein Spezialfall von B) ("is-a") (Bsp: B="reading", A="reading Shakespeare"
- Aggregationsbeziehungen (A ist Teil von B) ("part-of") (Bsp: B="car" A="wheel")
- Assoziationsbeziehungen (A hat allgemeinen Bezug zu B) (Bsp: A="Pinguin" B="Antarktis")
  etc.

Dies ist notwendigerweise eine **subjektive** Beurteilung, solle jedoch so sorgfältig wie möglich erfolgen.

3

	tv	movies	computers	women	drinking beer	skiing	dancing	sailing	cycling	cinema
inline-skating;										
playing drums;										
drawing;										
meeting friends;										
visiting exibitions;										
reading books;										
spead reading;										
party;										
playing cards and board games;										
icq'ing;										

4

Figure C.8: The first page and the third page of the questionnaire of survey 3

Users	Elements $\overset{\text{compared with}}{\longleftrightarrow}$ elements. (Numbering according to Figures C.1, C.2)			
	11 - 15 $\leftrightarrow$ 1 - 5	16 - 20 $\leftrightarrow$ 1 - 5	11 - 15 $\leftrightarrow$ 6 - 10	16 - 20 $\leftrightarrow$ 6 - 10
1	5 6 2 2 3	5 4 6 8 7	0 2 3 1 3	7 8 8 7 6
	3 2 4 2 1	4 6 5 6 6	8 7 6 2 2	7 5 6 3 3
	4 4 3 6 2	6 5 6 5 3	0 3 4 0 4	1 2 6 2 2
	2 1 5 2 2	2 6 4 5 4	7 8 9 3 2	0 1 3 2 8
	1 2 5 2 2	5 6 6 5 6	7 5 6 3 1	0 3 4 4 6
2	2 4 2 2 0	2 3 1 3 2	0.5 1 3 0.5 1	1 6 4 2 4
	2 0.5 1 4 3	1 0.5 0.5 2 1	4 8 4 3 3	1 3 3 1 2
	6 3 3 2 0.5	1 2 4 1 1	0 1 0.5 0.5 1	0.5 2 4 1 1
	2 1 1 3 1	0 1 0 1 1	3 1 4 1 1	1 1 3 1 2
	1 0.5 2 3 1	0 1 2 1 2	0.5 3 3 1 1	0.5 3 5 4 1
3	2 6 1 2 1	2 2 2 3 1	2 2 4 0 1	7 7 5 3 3
	3 2 1 2 3	1 3 2 3 1	3 7 4 2 3	6 3 6 1 3
	3 2 2 6 0	2 2 3 2 2	0 0 1 1 2	2 3 3 1 2
	2 3 2 3 1	1 3 0 2 2	1 2 5 1 1	2 2 8 2 1
	3 1 2 2 1	2 2 3 4 4	2 6 2 4 1	2 2 4 4 1
4	0 5 3 3 1	6 7 5 8 4	2 4 1 3 4	5 4 8 9 6
	5 0 4 6 4	3 3 4 4 3	5 8 7 7 6	6 5 6 4 4
	3 3 5 7 4	4 2 4 3 2	4 4 3 3 3	6 8 8 7 4
	3 3 4 2 2	0 1 3 3 4	8 7 6 4 3	7 5 4 7 4
	6 4 3 3 2	4 5 3 2 4	6 5 9 7 4	7 5 3 2 2
5	1 2 2 2 1	5 5 3 6 2	1 1 2 3 1	1 6 6 4 3
	8 5 8 3 5	1 2 2 3 1	6 8 5 3 6	2 2 2 2 2
	7 4 4 4 4	1 3 3 2 4	2 2 2 5 4	2 2 3 2 2
	2 1 2 7 3	1 3 1 2 2	3 3 3 4 2	2 2 2 2 3
	1 1 1 2 2	1 3 3 2 2	8 8 7 6 3	2 4 5 4 1
6	4 7 4 5 2	4 4 3 3 3	2 3 2 2 4	2 6 8 3 5
	4 3 3 5 4	2 4 5 4 3	6 7 4 4 4	2 3 4 2 3
	4 4 5 4 2	3 3 4 3 3	1 2 3 2 2	3 2 4 3 5
	4 4 3 2 1	1 6 2 3 4	3 4 4 3 2	2 3 6 3 4
	3 5 4 3 4	3 4 6 5 5	3 5 5 4 3	2 3 4 5 4
7	1 4 2 2 0	1 8 2 3 4	1 2 5 0 4	7 6 4 2 3
	2 3 4 7 4	1 1 4 3 2	0 5 3 6 1	2 1 3 1 2
	7 4 2 2 0	2 1 4 4 2	0 3 3 3 3	2 2 5 5 2
	3 5 5 1 0	0 1 0 5 5	1 4 5 3 2	1 1 5 3 6
	2 1 2 2 0	1 3 4 5 4	0 8 4 4 2	1 2 4 6 3
8	2 5 1 1 0	1 3 1 3 1	1 1 3 0 1	4 5 3 3 4
	2 2 3 3 3	1 3 3 2 0	2 6 3 5 2	6 3 5 1 3
	2 1 2 2 0	1 3 3 2 1	0 1 1 1 0	1 2 4 1 2
	2 2 1 0 0	0 3 0 4 2	2 1 7 3 2	2 2 4 1 1
	2 2 1 2 1	1 2 2 2 3	2 4 2 2 1	1 2 5 6 2

Table C.2: The Results (ratings $r_u(j_1, j_2)$) of Survey 1

Users	Elements $\overset{\text{compared with}}{\longleftrightarrow}$ elements. (Numbering according to Figures C.3, C.4)			
	11 - 15 $\leftrightarrow$ 1 - 5	16 - 20 $\leftrightarrow$ 1 - 5	11 - 15 $\leftrightarrow$ 6 - 10	16 - 20 $\leftrightarrow$ 6 - 10
1	7 9 9 9 8	7 9 9 6 9	7 8 7 6 6	7 9 8 8 7
	8 9 9 9 9	8 8 8 7 9	6 8 5 8 9	8 8 6 8 8
	8 10 8 9 8	9 9 7 6 10	4 8 2 8 4	7 7 7 7 8
	10 10 7 8 9	8 8 7 8 10	3 6 8 7 4	6 7 5 4 5
	8 9 5 3 10	9 9 6 8 8	6 8 5 6 4	5 10 0 2 5
2	7 9 7 6 8	6 5 7 6 7	6 6 5 8 5	7 8 6 6 7
	7 6 8 6 7	7 5 8 6 7	9 6 6 6 8	8 7 5 7 6
	8 6 7 7 7	4 4 5 5 4	6 5 6 5 4	5 6 4 5 5
	6 7 5 6 7	6 6 5 8 6	6 5 4 5 4	7 5 6 5 5
	4 4 6 4 5	5 4 6 5 5	3 5 1 2 5	7 5 4 5 5
3	5 7 4 5 6	9 8 7 5 8	4 7 1 6 3	7 5 1 4 6
	5 6 5 6 8	7 8 5 6 9	6 8 3 5 7	8 5 2 4 6
	5 8 5 6 8	7 4 6 5 8	3 4 5 2 3	0 5 1 2 3
	8 9 5 5 8	4 6 5 8 7	1 4 2 3 2	2 3 5 4 1
	2 6 2 0 9	7 8 6 5 4	6 8 0 4 5	3 6 2 3 3
4	7 8 10 6 6	5 9 8 6 5	5 10 8 5 7	6 9 5 6 5
	2 4 5 9 6	8 9 6 6 7	8 9 6 8 9	9 9 8 6 5
	6 10 8 9 5	8 5 4 4 10	6 4 6 4 6	6 5 8 4 7
	9 8 8 6 8	8 4 8 9 10	8 8 5 5 7	3 6 5 7 4
	4 9 8 5 10	9 8 6 8 9	8 10 1 3 5	2 7 6 7 2
5	9 7 7 7 7	6 7 6 6 7	6 9 3 7 6	4 6 1 4 4
	6 6 6 9 7	6 6 6 7 6	4 6 1 7 7	6 6 4 7 7
	6 7 6 7 7	4 4 1 3 4	4 4 4 6 6	3 4 1 3 4
	7 6 4 6 7	4 6 4 6 6	4 6 3 4 4	3 4 4 6 3
	4 4 4 3 6	4 4 1 3 3	4 6 1 3 3	3 4 3 3 1
6	7 7 5 6 6	5 6 6 4 8	4 7 2 7 4	5 5 2 4 4
	5 4 7 7 7	6 5 7 7 8	6 5 3 7 8	7 6 5 8 7
	5 5 5 7 6	6 4 4 3 5	4 4 3 5 6	3 4 3 3 4
	7 5 6 4 6	6 4 5 8 6	4 4 3 3 4	3 3 6 4 2
	1 2 3 0 4	7 5 4 4 3	2 4 0 1 3	4 4 3 3 3
7	8 7 6 7 6	7 6 5 4 6	6 5 2 5 2	8 7 3 5 4
	6 6 8 7 8	5 6 5 8 4	6 4 4 8 8	8 8 6 8 5
	5 8 5 7 7	4 3 5 2 4	5 6 8 3 3	2 3 4 4 6
	5 7 5 6 7	7 4 7 6 5	3 6 4 2 1	3 4 7 3 2
	3 5 3 0 6	6 3 5 3 4	2 4 0 1 2	3 5 3 3 2
8	9 8 7 7 8	8 9 9 8 10	9 7 6 8 6	6 7 6 4 5
	7 8 8 7 10	7 8 9 8 9	7 7 6 6 8	8 9 7 8 7
	7 10 8 9 9	9 9 7 7 10	6 7 9 4 7	6 8 6 4 5
	9 9 9 8 8	8 8 7 8 10	6 7 7 6 5	6 6 9 7 6
	8 9 7 3 9	9 9 7 7 7	5 8 0 3 4	5 8 6 6 4
9	7 8 3 8 9	7 6 7 8 7	5 9 7 9 8	8 5 6 7 7
	6 7 7 7 8	7 7 7 7 7	8 7 6 7 7	7 7 6 9 8
	4 8 5 8 7	6 4 8 6 8	6 6 8 6 8	6 6 6 6 7
	9 6 8 6 7	6 6 7 6 8	6 5 5 5 5	6 6 8 7 7
	9 7 7 6 8	5 8 6 6 7	5 7 4 4 4	8 7 6 6 6

Table C.3: The Results (ratings $r_u(j_1, j_2)$) of Survey 2

Users	Elements $\overset{\text{compared with}}{\longleftrightarrow}$ elements. (Numbering according to Figure C.5)			
	11 - 15 ↔ 1 - 5	16 - 20 ↔ 1 - 5	11 - 15 ↔ 6 - 10	16 - 20 ↔ 6 - 10
1	0 5 1 4 7 5 0 0 4 10 3 0 1 0 0 0 0 6 5 2 7 0 4 4 0 0 0 4 2 1 5 6 6 6 6 4 0 6 7 7 5 0 0 0 4 3 0 5 5 3 9 0 4 6 0 0 4 3 3 0 5 0 6 2 0 0 7 0 5 0 2 5 3 5 4 3 0 6 0 4 1 0 4 5 0 0 0 0 4 0 5 0 10 0 0 0 0 0 6 0	3 2 1 4 1 0 2 4 1 5 0 1 1 0 5 1 1 1 1 8 4 4 1 7 3 2 0 1 0 6 2 0 1 5 0 1 0 2 0 1 6 5 4 6 0 0 4 6 6 0 2 3 4 4 1 1 1 3 2 1 3 0 4 0 0 0 3 2 0 0 2 3 3 6 2 0 6 7 2 3 0 0 0 1 4 2 0 1 3 1 4 1 6 0 0 0 0 5 3 0	2 2 1 4 0 4 4 2 3 0 2 3 1 5 5 2 3 2 3 2 3 2 4 3 2 0 4 3 0 2 3 3 6 7 4 4 6 5 2 6 2 2 0 5 2 1 1 1 1 1 1 2 4 3 0 0 3 6 1 3 0 0 6 0 0 0 2 0 2 0 2 1 3 8 5 4 6 4 2 0 3 2 4 3 2 0 0 0 0 0 0 3 10 6 0 0 0 0 0 3	2 2 1 3 3 0 0 0 0 2 0 0 1 4 5 0 0 0 0 0 1 1 1 4 2 1 1 3 4 2 0 1 1 4 0 0 0 0 0 0 5 4 4 2 4 0 1 0 0 4 4 5 4 2 2 3 1 2 0 2 2 1 4 0 0 0 0 0 0 1 4 3 3 5 3 0 0 0 0 4 0 1 0 5 0 7 4 2 6 1 1 2 6 3 3 0 0 0 0 1
2	0 7 1 1 7 3 0 0 0 5 0 0 1 0 0 0 0 6 0 2 1 0 3 0 0 0 0 1 0 0 0 3 5 4 4 3 0 3 2 5 3 0 2 0 0 0 1 0 0 0 10 0 4 4 0 0 4 0 0 0 9 0 3 4 0 0 8 2 0 0 0 7 5 2 0 3 0 7 1 0 3 5 1 3 0 0 0 1 0 0 7 0 8 0 0 0 0 2 2 0	0 1 1 0 1 0 1 4 1 3 0 3 0 0 4 0 0 0 0 8 0 4 0 4 2 0 0 0 0 7 0 1 0 0 5 0 0 1 2 4 4 0 2 7 0 4 2 4 1 0 1 0 1 1 0 4 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 6 0 3 1 1 0 0 0 1 0 0 6 0 0 0 1 3 3 0 0 0 0 1 3 4 0 0	1 0 1 2 3 5 1 2 3 0 0 0 1 0 3 0 0 0 0 0 2 2 3 2 0 0 0 0 0 0 3 4 5 4 7 3 4 4 3 6 0 0 2 4 0 0 0 0 0 0 0 0 4 3 0 0 0 0 0 0 0 0 3 3 0 0 0 0 0 0 0 1 5 4 10 3 8 0 0 2 0 1 1 1 6 0 0 0 0 0 0 0 8 5 5 0 0 0 0 0	4 0 1 4 4 1 0 0 1 0 8 0 0 3 5 1 0 1 2 0 5 0 0 3 1 1 0 5 1 0 1 0 0 0 0 1 0 1 1 0 5 5 2 1 5 0 0 0 0 2 6 4 1 4 3 0 0 0 0 2 0 0 0 3 0 0 0 0 0 0 5 3 0 5 0 0 0 0 0 1 1 0 0 1 0 7 0 0 0 0 3 0 0 2 0 0 0 0 0 0
3	0 5 0 0 6 5 0 0 2 10 0 0 3 0 0 0 0 10 0 2 5 0 5 0 0 0 4 0 0 0 0 0 3 7 2 5 0 4 2 3 3 0 3 0 0 0 2 0 0 0 9 0 4 8 0 0 8 0 0 0 9 0 3 4 0 0 8 2 0 0 0 2 0 5 0 3 0 9 0 0 1 0 7 6 0 0 0 0 0 0 8 0 9 0 0 0 3 0 2 0	0 0 7 0 5 0 0 6 5 8 0 2 6 0 8 0 0 0 0 10 0 6 0 8 2 2 0 0 0 10 0 1 0 0 6 2 0 0 1 8 5 4 0 9 0 0 0 4 1 0 0 0 0 2 0 0 0 0 0 0 2 0 9 0 0 0 0 0 4 0 1 0 0 8 0 0 0 2 0 0 0 0 0 1 7 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	1 0 0 1 0 5 2 2 6 0 1 1 3 1 4 0 7 0 0 1 3 5 5 1 0 0 0 0 0 6 2 6 3 8 9 5 8 0 2 8 7 0 3 0 0 0 0 0 0 0 4 4 4 0 0 0 0 0 0 4 0 0 3 3 0 0 0 0 0 0 4 6 0 8 9 1 9 0 0 2 0 0 4 1 3 0 0 0 0 0 0 0 9 1 0 0 0 0 0 0	8 6 5 0 0 0 0 0 6 5 8 4 5 0 2 1 0 1 7 2 2 0 0 0 0 8 0 3 6 0 1 0 0 0 0 6 7 2 5 0 9 7 8 1 8 0 2 0 0 6 7 3 0 5 9 0 0 0 0 0 0 0 7 5 0 0 0 0 0 4 7 4 0 10 8 0 0 0 0 2 3 1 0 0 0 9 0 0 8 0 0 0 0 0 0 0 0 0 0 0
4	0 0 0 0 0 0 0 0 0 5 0 0 0 0 0 0 0 5 0 0 0 0 3 0 0 0 0 0 0 0 0 0 3 0 3 0 0 0 5 3 3 0 0 0 0 0 3 0 0 0 10 0 3 3 0 0 5 0 0 0 9 0 3 4 0 0 8 2 0 0 0 0 0 0 0 0 0 5 0 0 0 0 0 3 0 0 0 3 0 0 7 0 8 0 0 0 0 2 2 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 9 0 5 0 5 0 0 0 0 0 9 9 0 0 0 3 0 0 0 0 9 3 0 0 9 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 3 0 0 0 3 0 0 0 0 0 0 3 0 0 0 0 0 0 0 0 0 0 5 0 0 0 0 3 3 7 3 3 3 7 0 3 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 3 0 0 0 0 0 3 0 0 0 0 0 0 0 0 3 3 3 3 3 3 3 3 3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 3 3 0 0 0 0 0 0 0 0 0 3 3 0 3 0 0 0 0 0 0 3 0 0 0 0 0 0 0 0 8 5 5 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 3 0 3 0 0 0 0 0 0 0 0 0 0 0 0 0 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 7 0 0 3 0 0 0 0 0 0 0 0 0 0 0
5	0 5 0 0 5 5 0 0 0 10 0 0 0 0 0 0 0 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 10 0 0 0 0 0 5 0 0 0 10 0 0 0 0 0 5 0 0 0 0 0 0 0 0 0 0 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 5 0 0 0 0 0 5 0	0 0 0 0 0 0 0 0 0 0 5 0 0 0 0 5 5 0 0 0 5 0 5 0 5 0 5 0 0 0 10 0 5 0 0 5 5 0 0 0 10 0 0 0 10 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 5 0 0 0 0 0	0 0 0 0 0 5 5 5 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 5 5 5 5 0 0 0 0 0 0 5 5 5 5 0 0 0 0 0 0 5 5 5 5 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
6	0 8 0 4 8 8 0 0 0 10 3 4 4 0 4 4 0 9 0 4 6 4 4 2 4 2 2 6 0 4 1 3 4 7 6 6 1 7 9 4 5 3 2 8 3 2 6 5 3 2 10 0 8 4 0 0 9 6 0 0 10 0 6 0 0 0 7 1 0 0 1 3 0 8 3 4 1 9 9 4 4 3 8 6 3 0 3 0 0 5 7 0 10 0 0 0 3 0 7 0	0 0 2 0 6 0 5 4 0 5 0 5 9 0 7 0 2 1 0 8 0 8 0 7 4 4 2 0 0 8 0 6 0 4 9 4 2 0 0 8 5 4 5 8 3 3 6 3 5 2 4 0 2 3 1 1 0 0 0 3 4 0 3 0 0 0 2 0 0 0 0 0 0 8 0 0 0 3 2 1 0 0 0 4 5 0 0 0 0 7 5 3 7 0 0 0 2 8 5 2	0 0 0 2 0 7 5 5 7 0 0 0 5 1 3 3 5 1 1 1 2 2 5 4 0 1 3 1 1 2 2 4 4 8 8 5 6 5 4 5 4 4 2 5 3 1 5 3 0 4 4 1 3 2 1 0 1 0 0 2 4 1 3 2 1 0 1 0 0 2 2 4 1 8 9 3 7 2 0 5 4 5 6 4 5 2 2 2 1 2 0 3 9 7 0 0 0 0 0 0	4 4 4 0 4 4 0 1 4 2 5 3 3 0 3 6 5 5 7 2 1 0 0 5 0 7 5 5 5 2 1 0 0 3 0 7 5 5 7 2 7 4 6 4 6 0 2 2 2 5 5 4 0 4 9 1 0 0 2 2 0 0 2 1 0 0 0 0 0 2 2 2 0 7 2 0 0 0 0 2 2 4 0 3 2 9 0 3 5 3 4 5 6 8 8 2 4 1 0 5
7	0 7 0 3 8 5 0 0 0 10 3 0 0 3 0 0 0 10 0 0 5 0 5 2 0 0 0 0 0 0 0 7 3 10 7 7 0 7 7 7 0 0 0 0 0 0 0 0 0 0 10 0 3 8 0 0 7 5 0 0 10 0 6 0 0 0 7 1 0 0 0 5 2 7 3 5 0 10 0 5 0 0 3 5 0 0 0 0 0 0 8 0 9 0 0 0 3 0 2 0	0 5 4 0 4 0 0 0 0 5 0 6 0 0 7 3 0 0 0 10 0 8 0 5 7 7 0 0 0 10 0 5 0 0 6 5 0 0 0 10 8 0 3 10 0 0 5 5 3 0 0 0 0 0 0 0 0 0 0 0 0 0 5 0 0 0 0 0 0 0 0 0 0 3 0 0 0 0 0 0 0 0 0 3 0 0 0 0 0 3 0 0 0 0 0 0 0 3 0 0	0 0 0 0 0 5 4 4 5 0 0 0 0 0 0 6 0 5 0 0 0 0 0 5 0 0 0 0 0 0 0 3 5 3 4 8 7 7 7 7 7 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 3 3 0 0 0 0 0 0 0 0 0 5 8 0 8 0 0 0 0 0 3 0 3 0 0 0 0 0 0 0 8 5 5 0 0 0 0 0	0 0 4 0 0 3 0 3 4 0 0 0 0 4 0 5 0 3 7 0 0 0 0 0 0 5 4 5 5 0 0 0 0 0 0 5 4 5 5 0 8 8 8 0 7 0 0 0 0 5 5 3 3 4 6 0 0 0 0 0 0 0 5 0 0 0 0 0 0 0 0 0 0 5 0 0 0 0 0 0 0 0 0 0 0 8 0 0 3 0 3 3 0 0 0 0 0 0 0 3

Table C.4: The Results (ratings $r_u(j_1, j_2)$) of Survey 3

Survey 1

Elements $\overset{\text{compared with}}{\longleftrightarrow}$ elements. (Numbering according to Figures C.1, C.2)

	11 - 15 ↔ 1 - 5					16 - 20 ↔ 1 - 5				
$Av(j_1, j_2)$	2.125	4.875	2.125	2.375	1.0	3.25	4.5	2.875	4.625	3.0
	3.625	2.1875	3.5	4.0	3.375	1.75	2.8125	3.1875	3.375	2.125
	4.5	3.125	3.25	4.125	1.5625	2.5	2.625	3.875	2.75	2.25
	2.5	2.5	2.875	2.5	1.25	0.625	3.0	1.25	3.125	3.0
	2.375	2.0625	2.5	2.375	1.625	2.125	3.25	3.625	3.25	3.75
$\sigma(j_1, j_2)$	1.642	1.5526	0.991	1.1877	1.069	1.982	2.0701	1.8077	2.326	2.0
	2.0658	1.5569	2.2038	1.8516	1.1877	1.1649	1.7307	1.6021	1.3024	1.885
	1.9272	1.1259	1.2817	2.031	1.7204	1.7728	1.1877	0.991	1.2817	1.035
	0.7559	1.5118	1.642	2.0701	1.035	0.744	2.0701	1.5811	1.4577	1.4142
	1.685	1.6132	1.4142	0.5175	1.1877	1.7268	1.669	1.5979	1.669	1.3887

Survey 2

Elements $\overset{\text{compared with}}{\longleftrightarrow}$ elements. (Numbering according to Figures C.3, C.4))

	11 - 15 ↔ 1 - 5					16 - 20 ↔ 1 - 5				
$Av(j_1, j_2)$	7.3333	7.7777	6.4444	6.7777	7.1111	6.6666	7.2222	7.1111	5.8888	7.4444
	5.7777	6.2222	7.0	7.4444	7.7777	6.7777	6.8888	6.7777	6.8888	7.3333
	6.0	8.0	6.3333	7.6666	7.1111	6.3333	5.1111	5.2222	4.5555	7.0
	7.7777	7.4444	6.3333	6.1111	7.4444	6.3333	5.7777	6.1111	7.4444	7.5555
	4.7777	6.1111	5.0	2.6666	7.4444	6.7777	6.4444	5.2222	5.4444	5.5555
$\sigma(j_1, j_2)$	1.2247	0.8333	2.2422	1.2018	1.1666	1.3228	1.5634	1.3642	1.4529	1.5092
	1.7159	1.6414	1.4142	1.236	1.2018	0.9718	1.4529	1.3944	0.7817	1.6583
	1.4142	1.8027	1.4142	1.118	1.1666	2.0615	2.2607	2.1081	1.6666	2.7386
	1.6414	1.6666	1.732	1.2692	0.8819	1.5811	1.5634	1.3642	1.1303	2.0069
	2.8625	2.5712	2.1213	2.236	2.2422	1.922	2.4037	1.7873	1.9436	2.2422

Survey 3

Elements $\overset{\text{compared with}}{\longleftrightarrow}$ elements. (Numbering according to Figure C.5)

	21 - 30 ↔ 1 - 10										31 - 40 ↔ 1 - 10									
$Av(j_1, j_2)$	0.0	5.28	0.28	1.71	5.85	4.42	0.0	0.0	0.85	8.57	0.42	1.14	2.14	0.57	2.42	0.0	1.14	2.57	1.0	3.74
	1.28	0.57	1.28	0.42	0.57	0.57	0.0	7.28	0.71	1.42	0.71	2.42	2.28	0.0	5.14	1.28	0.42	0.28	0.14	8.28
	3.42	0.57	3.42	1.14	0.57	0.28	0.85	1.57	0.28	0.71	0.57	5.71	0.14	5.85	2.57	2.85	0.28	0.14	0.0	8.57
	0.85	2.71	3.42	4.85	4.0	3.57	0.14	3.85	4.57	4.14	1.57	2.57	0.14	1.28	4.85	2.42	0.28	0.42	0.42	7.14
	2.71	0.42	1.0	1.14	1.0	0.71	1.71	1.42	1.14	0.71	4.42	1.85	2.0	8.42	0.42	1.0	2.42	3.14	2.28	0.28
	9.71	0.0	3.71	4.71	0.0	0.0	6.0	2.0	0.42	0.0	1.0	0.42	1.0	1.42	0.28	0.85	0.14	0.42	0.28	0.71
	8.85	0.0	3.85	2.0	0.0	0.0	7.14	1.14	0.71	0.0	1.28	0.0	3.42	0.0	0.0	0.0	1.14	0.28	0.57	0.0
	0.42	3.14	1.42	3.85	1.42	2.57	0.14	7.28	1.42	1.85	0.42	0.42	0.42	5.57	0.28	0.42	1.0	1.85	0.57	0.57
	1.28	1.14	3.28	4.0	0.42	0.0	0.42	0.57	0.57	0.71	0.0	0.14	0.0	1.28	3.85	0.28	0.0	0.14	0.57	2.42
	6.0	0.0	8.42	0.0	0.0	0.0	1.28	0.57	3.71	0.0	2.14	1.57	2.28	0.42	0.42	1.14	0.71	4.0	1.14	0.28
$\sigma(j_1, j_2)$	0.0	2.62	0.48	1.88	2.79	2.43	0.0	0.0	1.57	2.43	1.13	1.86	2.54	1.51	2.50	0.0	1.86	2.50	1.82	2.92
	1.60	1.51	1.60	1.13	1.51	1.51	0.0	2.28	1.88	1.51	1.88	2.37	3.68	0.0	2.67	1.97	0.78	0.48	0.37	1.70
	2.9	1.51	1.71	1.57	1.51	0.75	1.57	2.43	0.75	1.49	1.51	1.70	0.37	1.46	2.43	2.60	0.75	0.37	0.0	1.61
	1.86	2.92	1.90	3.76	2.51	2.76	0.37	3.02	3.30	2.47	3.35	2.63	0.37	2.21	2.79	2.22	0.75	0.78	0.78	3.38
	2.05	1.13	1.29	3.02	1.73	1.25	2.21	2.43	2.03	1.25	2.50	2.34	2.08	1.51	1.13	1.73	2.57	2.34	2.42	0.75
	0.48	0.0	2.36	2.87	0.0	0.0	2.0	2.64	1.13	0.0	1.52	1.13	1.52	1.61	0.48	1.46	0.37	1.13	0.75	1.11
	1.77	0.0	2.26	2.0	0.0	0.0	1.06	0.89	1.88	0.0	1.70	0.0	3.10	0.0	0.0	0.0	1.46	0.755	1.51	0.0
	0.78	2.67	1.98	3.23	1.81	1.90	0.37	2.05	3.35	2.34	0.78	1.13	1.13	2.07	0.75	1.13	2.23	2.54	0.97	1.13
	1.60	2.03	3.25	2.16	1.13	0.0	1.13	1.13	1.51	1.88	0.0	0.37	0.0	1.60	2.79	0.75	0.0	0.37	1.13	2.43
	2.82	0.0	1.71	0.0	0.0	0.0	1.60	0.97	2.21	0.0	2.11	2.63	3.09	1.13	1.13	2.60	1.25	2.44	2.03	0.75

Table C.5: The Average Results and Standard Deviations of Survey 1, 2 and 3 (Part I). $\sigma(j_1, j_2)$ denotes Standard deviation ($\sigma(j_1, j_2) = \sqrt{\frac{1}{7}\sum_{u=1}^{8}(r_u(j_1, j_2) - Av(j_1, j_2))^2}$). $Av(j_1, j_2)$ denotes Average rating ($Av(j_1, j_2) = \frac{1}{8}\sum_{u=1}^{8} r_u(j_1, j_2)$

Survey 1																				
	Elements $\overset{\text{compared with}}{\longleftrightarrow}$ elements. (Numbering according to Figures C.1, C.2)																			
	11 - 15 $\leftrightarrow$ 6 - 10										16 - 20 $\leftrightarrow$ 6 - 10									
$Av(j_1, j_2)$	1.1875	2.0	2.875	1.1875	2.375						4.25	6.0	5.75	4.125	4.25					
	4.25	7.0	4.5	4.0	3.375						4.0	3.125	4.375	1.875	2.75					
	0.875	2.0	2.1875	1.9375	2.375						2.1875	2.875	4.625	2.75	2.5					
	3.5	3.75	5.375	2.75	1.875						2.125	2.125	4.375	2.625	3.625					
	3.5625	5.5	4.75	3.875	2.0						1.9375	3.0	4.25	4.375	2.5					
$\sigma(j_1, j_2)$	0.7529	1.069	1.2464	1.3076	1.5059						2.6592	1.1952	2.0528	2.5319	1.2817					
	2.5495	1.069	1.4142	1.8516	1.8468						2.4494	1.3562	1.5979	1.1259	0.7071					
	1.4577	1.3093	1.2517	1.6569	1.4078						1.7307	2.1001	1.685	2.1876	1.3093					
	2.6186	2.6049	1.9226	1.1649	0.6408						2.1001	1.3562	1.9226	1.9226	2.4458					
	3.0406	1.7728	2.4928	1.9594	1.1952						2.1784	1.069	0.7071	1.3024	1.7728					

Survey 2																				
	Elements $\overset{\text{compared with}}{\longleftrightarrow}$ elements. (Numbering according to Figures C.3, C.4))																			
	11 - 15 $\leftrightarrow$ 6 - 10										16 - 20 $\leftrightarrow$ 6 - 10									
$Av(j_1, j_2)$	5.7777	7.5555	4.5555	6.7777	5.2222						6.4444	6.7777	4.2222	5.3333	5.4444					
	6.6666	6.6666	4.4444	6.8888	7.8888						7.6666	7.2222	5.4444	7.2222	6.5555					
	4.8888	5.3333	5.6666	4.7777	5.2222						4.2222	5.3333	4.4444	4.2222	5.4444					
	4.5555	5.6666	4.5555	4.4444	4.0						4.3333	4.8888	6.1111	5.2222	3.8888					
	4.5555	6.6666	1.3333	3.0	3.8888						4.4444	6.2222	3.6666	4.2222	3.4444					
$\sigma(j_1, j_2)$	1.5634	1.5898	2.6034	1.3944	1.922						1.3333	1.6414	2.5385	1.5	1.3333					
	1.5	1.5811	1.8104	1.054	0.7817						0.866	1.3944	1.74	1.4813	1.1303					
	1.1666	1.5	2.3979	1.7873	1.7873						2.3333	1.5811	2.5055	1.5634	1.6666					
	2.1278	1.3228	1.9436	1.5898	1.732						1.8708	1.4529	1.6158	1.5634	2.0275					
	2.0069	2.0615	1.8708	1.5811	1.054						2.0069	1.986	2.0615	1.7873	1.6666					

Survey 3																				
	Elements $\overset{\text{compared with}}{\longleftrightarrow}$ elements. (Numbering according to Figure C.5)																			
	21 - 30 $\leftrightarrow$ 11 - 20										31 - 40 $\leftrightarrow$ 11 - 20									
$Av(j_1, j_2)$	0.57	0.28	0.28	1.28	0.42	4.42	3.0	2.85	4.14	0.0	2.57	1.71	2.14	1.0	1.57	1.14	0.0	0.57	2.14	1.28
	0.42	0.57	1.42	1.0	3.0	0.71	3.28	0.42	0.57	0.57	3.0	1.0	1.28	1.57	2.14	1.85	0.71	2.14	4.0	0.57
	1.42	1.57	3.57	1.42	0.28	0.14	1.0	0.57	0.14	1.42	1.28	0.14	0.14	1.71	0.42	3.85	2.14	3.71	3.71	0.57
	1.85	3.57	3.42	4.85	5.57	3.85	4.85	3.42	3.0	5.0	0.42	0.14	0.14	1.0	0.0	3.42	3.0	2.57	3.71	0.28
	1.85	0.85	1.0	2.0	0.71	0.28	0.85	0.57	0.14	0.71	5.28	4.0	4.0	1.14	4.42	0.0	0.71	0.28	0.28	3.14
	1.28	1.0	2.14	1.14	0.14	0.0	0.57	0.85	0.14	1.28	3.85	2.71	1.14	2.71	5.0	0.57	0.14	0.28	0.28	0.85
	0.57	0.14	3.0	2.0	0.14	0.0	0.42	0.0	0.28	0.28	0.28	0.14	2.57	1.28	0.0	0.0	0.0	0.0	0.0	1.0
	1.14	1.71	1.28	5.85	7.0	1.57	5.85	0.85	0.28	1.28	2.57	1.71	0.42	4.57	1.85	0.0	0.0	0.0	0.0	1.28
	1.0	1.14	2.57	1.28	3.14	0.28	0.28	0.28	0.14	0.28	0.85	0.85	0.0	1.28	0.28	6.71	0.57	0.71	3.57	0.57
	0.0	0.85	7.42	4.14	2.14	0.0	0.0	0.0	0.0	0.42	1.57	1.42	1.71	1.85	1.57	0.28	0.57	0.14	0.0	1.42
$\sigma(j_1, j_2)$	0.78	0.75	0.48	1.49	1.13	2.14	2.0	1.86	2.34	0.0	2.99	2.42	2.11	1.73	1.98	1.67	0.0	1.13	2.47	1.88
	0.78	1.13	1.90	1.82	2.30	1.25	2.62	0.78	1.13	0.78	3.87	1.73	1.97	1.98	2.26	2.54	1.88	2.19	3.26	0.97
	1.39	1.81	1.81	1.61	0.75	0.37	1.73	1.13	0.37	2.22	1.79	0.37	0.37	2.21	0.78	3.18	2.41	1.88	2.28	0.97
	1.34	1.90	1.90	2.96	3.30	2.19	2.73	2.63	2.16	2.70	0.53	0.37	0.37	1.73	0.0	2.99	2.94	2.37	2.49	0.75
	2.73	1.57	1.29	2.51	1.25	0.48	1.86	1.13	0.37	1.49	3.09	3.10	3.46	1.46	2.99	0.0	0.95	0.75	0.75	2.47
	1.88	1.52	2.03	1.46	0.37	0.0	1.13	2.26	0.37	1.70	2.79	1.97	1.67	2.05	3.21	1.13	0.37	0.75	0.75	1.06
	1.51	0.37	1.73	1.41	0.37	0.0	0.78	0.0	0.75	0.75	0.75	0.37	2.81	1.9	0.0	0.0	0.0	0.0	0.0	1.52
	1.57	2.36	1.97	2.11	2.64	1.71	3.23	1.57	0.75	1.88	2.81	1.70	1.13	3.59	2.9	0.0	0.0	0.0	0.0	1.4
	1.73	1.86	2.29	1.60	1.95	0.75	0.75	0.75	0.37	0.75	1.21	1.46	0.0	1.9	0.75	3.09	1.51	1.25	2.9	1.13
	0.0	1.46	3.35	2.60	2.67	0.0	0.0	0.0	0.0	1.13	1.71	1.9	2.92	2.9	3.04	0.75	1.51	0.37	0.0	1.90

Table C.6: The Average Results and Standard Deviations of Survey 1, 2 and 3 (Part II). $\sigma(j_1, j_2)$ denotes Standard deviation ($\sigma(j_1, j_2) = \sqrt{\frac{1}{7}\sum_{u=1}^{8}(r_u(j_1, j_2) - Av(j_1, j_2))^2}$). $Av(j_1, j_2)$ denotes Average rating ($Av(j_1, j_2) = \frac{1}{8}\sum_{u=1}^{8} r_u(j_1, j_2)$)

Similarity Measure for sets of Free Text Interest Phrases defined in figures 4.9, 4.10 (compare Survey 1). Parameters were set to $\alpha = 0.7$ and $\beta = 0.3$.

Elements $\overset{\text{compared with}}{\longleftrightarrow}$ elements. (Numbering according to Figures C.1, C.2)

11 - 15 ↔ 1 - 5					16 - 20 ↔ 1 - 5				
0.2061	0.3604	0.1928	0.2397	0.1176	0.2101	0.2763	0.3591	0.3039	0.1089
0.2051	0.1637	0.3075	0.2781	0.2202	0.0759	0.3354	0.3288	0.2907	0.1110
0.0681	0.1471	0.3375	0.2612	0.1117	0.1962	0.3435	0.3573	0.3602	0.2296
0.0640	0.2463	0.1405	0.1414	0.1251	0.0396	0.2895	0.0781	0.1731	0.2188
0.2223	0.2080	0.2499	0.2095	0.0299	0.2266	0.3114	0.4438	0.4021	0.2003

Similarity Measure for List of Choice Interest Vectors defined in figure 4.5 (compare Survey 2)

Elements $\overset{\text{compared with}}{\longleftrightarrow}$ elements. (Numbering according to Figures C.3, C.4))

11 - 15 ↔ 1 - 5					16 - 20 ↔ 1 - 5				
0.7893	0.7032	0.7176	0.8510	0.4334	0.7335	0.7241	0.7239	0.6326	0.6928
0.7111	0.6490	0.7355	0.7430	0.5783	0.7369	0.6658	0.6379	0.5417	0.5886
0.7177	0.7626	0.6993	0.6976	0.2537	0.7500	0.8221	0.6526	0.7887	0.6465
0.7014	0.8093	0.8056	0.7022	0.2139	0.6536	0.7981	0.6509	0.8210	0.6176
0.7183	0.8036	0.7708	0.8039	0.6028	0.8117	0.7734	0.7339	0.7302	0.6168

Similarity Measure that compares single concept phrases from Sets of Free Text Interest Phrases. This similarity measure is defined in equation 4.12 and is a part of the similarity measure for Sets of Free Text Interest Phrases defined in figures 4.9, 4.10 (compare survey 3)

Elements $\overset{\text{compared with}}{\longleftrightarrow}$ elements. (Numbering according to Figure C.5)

21 - 30 ↔ 1 - 10										31 - 40 ↔ 1 - 10									
0.000	0.000	0.000	0.131	0.000	0.000	0.000	0.000	0.000	0.543	0.000	0.000	0.224	0.000	0.000	0.000	0.194	0.000	0.170	0.000
0.081	0.000	0.248	0.088	0.000	0.000	0.162	0.000	0.127	0.151	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.286	0.000	0.000	0.000	0.000	0.000	0.117	0.000	0.000	0.000	0.000	0.000	0.444
0.000	0.000	0.103	0.046	0.000	0.000	0.089	0.000	0.074	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.256	0.078	0.000	0.000	0.212	0.000	0.192	0.000
0.750	0.000	0.146	0.029	0.000	0.000	0.126	0.000	0.097	0.000	0.000	0.000	0.070	0.000	0.000	0.000	0.058	0.000	0.035	0.000
0.750	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.232	0.070	0.000	0.000	0.195	0.000	0.174	0.000
0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.067	0.000	0.000	0.090	0.000	0.000	0.000	0.000	0.000	0.171	0.000	0.000	0.135	0.000	0.000	0.000	0.116	0.000	0.099	0.000
0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Table C.7: The Results of the similarity measures which have been developed in chapter 4 and which correspond to Survey 1, 2 and 3 (Part I).

Similarity Measure for sets of Free Text Interest Phrases defined in figures 4.9, 4.10 (compare Survey 1). Parameters were set to $\alpha = 0.7$ and $\beta = 0.3$.

Elements $\overset{\text{compared with}}{\longleftrightarrow}$ elements. (Numbering according to Figures C.1, C.2)

11 - 15 ↔ 6 - 10					16 - 20 ↔ 6 - 10				
0.2325	0.2895	0.4103	0.1334	0.3036	0.3800	0.4787	0.2924	0.2913	0.3432
0.2309	0.3584	0.2695	0.3242	0.3106	0.4870	0.3388	0.3526	0.1728	0.3110
0.1415	0.1851	0.1681	0.1218	0.1204	0.2288	0.3227	0.3876	0.2020	0.3450
0.1825	0.2202	0.3079	0.1185	0.1900	0.2233	0.2080	0.3550	0.1467	0.1644
0.1677	0.4341	0.1884	0.2929	0.2684	0.2301	0.2761	0.3589	0.4164	0.2823

Similarity Measure for List of Choice Interest Vectors defined in figure 4.5 (compare Survey 2)

Elements $\overset{\text{compared with}}{\longleftrightarrow}$ elements. (Numbering according to Figures C.3, C.4))

11 - 15 ↔ 6 - 10					16 - 20 ↔ 6 - 10				
0.6350	0.7520	0.6834	0.6652	0.5034	0.6124	0.7852	0.5535	0.5078	0.6133
0.8353	0.6952	0.6907	0.6989	0.6512	0.6834	0.6940	0.6497	0.4793	0.6298
0.4693	0.5002	0.6742	0.4610	0.0000	0.4090	0.7337	0.4038	0.7447	0.4991
0.7629	0.7318	0.7207	0.6004	0.2042	0.6309	0.8909	0.5685	0.7127	0.5496
0.7123	0.8635	0.7801	0.6652	0.4705	0.6503	0.8075	0.7045	0.5436	0.4927

Similarity Measure that compares single concept phrases from Sets of Free Text Interest Phrases. This similarity measure is defined in equation 4.12 and is a part of the similarity measure for Sets of Free Text Interest Phrases defined in figures 4.9, 4.10 (compare survey 3)

Elements $\overset{\text{compared with}}{\longleftrightarrow}$ elements. (Numbering according to Figure C.5)

21 - 30 ↔ 11 - 20										31 - 40 ↔ 11 - 20									
0.000	0.000	0.000	0.000	0.000	0.444	0.340	0.175	0.444	0.000	0.194	0.000	0.224	0.093	0.018	0.000	0.000	0.000	0.000	0.224
0.162	0.000	0.248	0.085	0.044	0.137	0.217	0.088	0.137	0.191	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.000	0.286	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.375	0.290	0.156	0.375	0.000
0.089	0.000	0.103	0.300	0.054	0.000	0.000	0.000	0.000	0.103	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.212	0.000	0.256	0.147	0.083	0.000	0.000	0.000	0.000	0.256
0.126	0.000	0.146	0.085	0.044	0.000	0.088	0.000	0.000	0.146	0.058	0.000	0.070	0.091	0.180	0.000	0.000	0.000	0.000	0.070
0.000	0.000	0.000	0.000	0.000	0.000	0.088	0.000	0.000	0.000	0.195	0.000	0.232	0.135	0.077	0.000	0.000	0.000	0.000	0.232
0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.000	0.146	0.000	0.000	0.000	0.148	0.246	0.140	0.148	0.000	0.116	0.000	0.135	0.127	0.023	0.000	0.000	0.000	0.000	0.135
0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Table C.8: The Results of the similarity measures which have been developed in chapter 4 and which correspond to Survey 1, 2 and 3 (Part II).

Bibliography

[1] K. Aas and L. Eikvil. Text Categorisation: A Survey. Technical report, Norwegian Computing Center, Oslo, 1999.

[2] M. Ackerman et al. *Learning probabilistic user profiles* AI Magazine, 18(2) p. 47-56, 1997

[3] M. Adelwart, S Claus, C. Pfaller, C. Wied *Terminalbasierte Standortbestimmung in GSM Netzen: Enwicklung eines Vefahrens zur Positionsbestimmung* Systementwiockungsprojekt, Informatik XIII, Fakultät für Informatik, TU München, Feb. 2002

[4] G. Abowd, C. Atkeson, J. Hong, S. Long, R. Kooper, M. Pinkerton *Cyberguide - A Mobile Context-Aware Tour Guide* Wireless Networks, 3(5), pp. 421-433, Oct. 1997

[5] T. Allen *Managing the Flow of Technology* MIT Press, Cambridge, 1977

[6] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval.* ACM press, 2000.

[7] J. Bajada *Mobile Positioning for Location Dependent Services in GSM Networks.* Proc. CSAW 2003, University of Malta, 2003.

[8] U. Baumgarten, H. Krcmar, R. Reichwald, J. Schlichter *Community Online Services and Online Solutions - Projektstartbericht des Verbundvorhabens COSMOS.* Technical Report TUM-I0105, Munich University of Technology, Okt. 2001

[9] A. Bellebaum *Soziologische Grundbegriffe.* Verlag W. Kohlhammer, Stuttgart, 2001

[10] T. Berners-Lee, J. Hendler O. Lassila *The Semantic Web.* Scientific American, May 2001

[11] J. Bezdeck, R. Ehrlich, W. Full *FCM:Fuzzy C-Means Algorithm.* Computers and Geoscience, 1984

[12] N. Bolshakova, F. Azuaje *Improving Expression Data Mining Through Cluster Validation.* Proc. 4th Annual IEEE Conference on IT Applications in Biomedicine pp 19-22, 2003.

[13] E. Bradner *Computer Mediated Communication Among Teams: What are "Teams" and how are they "Virtual"?* in C. Lueg, D. Fisher (Ed.): From UseNet to CoWeb - Studying Social Information Spaces, Springer 2003

[14] E. Bradner, G. Mark *Why Distance Matters: Effects on Cooperation, Persuation and Deception* Proc. CSCW02, New Orleans, Nov. 2002

[15] (G. Booch, I. Jacobsen, J. Rumbaugh et al.) *OMG Unified Modeling Language Specification.* http://www.omg.org/docs/formal/03-03-01.pdf (URL, Jan. 2004)

[16] U. Borghoff, J. Schlichter, *Computer Supported Cooperative Work.*. Springer, 2000.

[17] J. Buckley and E. Eslami *Fuzzy Plane Geometry II: Circles and Polygons.* Fuzzy Sets and Systems 87(1997) pp.79-85

[18] A. Budanitzky *Lexical Semantic Relatedness and its Application in Natural Language Processing.* Tech. report CSRG-390, Computer Systems Research Group, University of Toronto Aug. 1999.

[19] A. Budanitzky, G. Hirst *Semantic Distance in WordNet: An experimental, application oriented evaluation of five measures.* Worshop on WordNet and Other Lexcial Resources, NAACL-2000, Pittsburgh, Jun. 2001

[20] H. Bullinger, T. Baumann, N. Fröschle, O. Mack, T. Trunzer, and J. Waltert. *Business Communities.* Galileo Press, Bonn, Germany, Jan. 2002.

[21] J. Breese, D. Heckerman, C. Kadie *Empirical Analysis of Predictive Algorithms for Collaborative Filtering.* Technical Report MSP-TR-98-12, Microsoft Research, 1998

[22] T. Camp, J. Boleng, V. Davies *A Survey of Mobility Models for Ad-Hoc-Network Research.* Wireless Communications and Mobile Computing (WCMC): Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications, nr. 5, vol. 2, p. 483-502, 2002

[23] L. Carotenuto, W. Etienne, M. Fontaine, J. Friedman, H. Newberg, M. Muller, M. Simpson, J. Slusher, K. Stevenson *CommunitySpace: Toward Flexible Support for Voluntary Knowledge Communities.* In Proc. Workshop on Workspace Models for Collaboration, London, UK, April 1999.

[24] J. Carroll, S. Laughton, M. Rosson, *Network Communities.* Proc. CHI'96 Conference on Human Factors in Computing Systems, p. 357f., Vancouver, Canada, 1996

[25] P. Chan *Non-Invasive Learning Approach to Building Web User Profiles.* Workshop on Web Usage Analysis and User Profiling, pp. 7-12, ACM Press, NY, 1999

[26] G. Chen, D. Kotz *A Survey of Context-Aware Mobile Computing Research.* Dartmouth Computer Science Technical Report TR2000-381, 2000.

[27] M. Claypool and D. Brown and M. Phong, *Inferring User Interests.* IEEE Internet Computing, November-December 2001

[28] N. Davies, K. Mitchell, K. Cheverest, G. Blair *Developing a Context Sensitive Tour Guide.* Tech. Report, Computing Department, Lancaster University, Mar. 1998

[29] J. Davies, D. Fensel, F. van Harmelen (Eds.) *Towards the Semantic Web - Ontology driven Knowledge Management.* John Wiley & Sons, 2003

[30] A. Dey *Understanding and Using Context.* Personal and Ubiquitous Computing, Vol. 5, 2001.

[31] N. Döring *Sozialpsychologie des Internet.* Hofgrefe, Verlag für Psychologie, 1999.

[32] J. Donath, K. Karahalios, F. Viegas *Visualizing Conversation.* Proc. Hawaii Int'l Conf. on System Sciences 32, Jan. 1999

[33] E. Dorrer *From Elliptic Arc Length to Gauss-Krüger Coordinates by Analytical Continuation* Technical Reports Department of Geodesy and Geoinformatics, University of Stuttgart, Report-Nr. 1999.6, p. 91–100, 1999

[34] E. Dyson *Release 2.0 - A Design for Living in the Digital Age.* Broadway Books, NY, 1997.

[35] C. Ehmig, M. Geiger, R. Friess, C. Hillebrand, G. Groh, R. Kerl, O. Brakel, A. Tasch *jetzt.de mobile community platform.* http://www.jetzt.de/pda/ (URL, Jan. 2004).

[36] G. Endruweit, G. Trommsdorff (Eds.) *Wörterbuch der Soziologie.* F. Enke Verlag, Stuttgart, 1989

[37] D. Etzold *Stability Behaviour of Fuzzy Clustering Methods for Text Data and Locations.* Diploma Thesis, TU-München, 2004, supervised by G. Groh

[38] C. Fellbaum *Introductory Remarks on WordNet* in C. Fellbaum (Ed.) - WordNet: an electronic lexical database, MIT Press, 1998

[39] L. Festinger *A Theory of Social Comparison Process* Human Relations 7, pp. 117-140, 1954

[40] L. Fischer, G. Wiswede *Grundlagen der Sozialpsychologie* Oldenbourg Verlag, 1997

[41] D. Fisher *Studying Social Information Spaces* in C. Lueg, D. Fisher (Ed.): From UseNet to CoWeb - Studying Social Information Spaces, Springer 2003

[42] E. Franconi *Description Logics* Lecture Slides, Http://www.inf.unibz.it/ franconi/dl/course/ (URL, Jan. 2004)

[43] N. Friedman-Noy, C. Hafner *The State of the Art in Ontology Design* AI Magazine, 18(3), 53-74, 1997

[44] J. Fink, A. Kobsa *A Review and Analysis of Commercial User Modeling Servers for Personalization on the World Wide Web* User Modeling and User-Adapted Interaction 10, pp. 209-249, 2000

[45] F. Fukomoto, Y. Suzuki, J. Fukomoto *An Automatic Extraction of Key Paragraphs Based on Context Dependency* in Proc. 5th Conf. on Applied Natural Language Processing, Washington, USA, pp. 291-298, 1997

[46] M. Galla *Interpersonal Relationship-Management in Shared Information Spaces* PhD thesis, Technical University of Munich, Mar. 2004

[47] A. Gelb *Applied Optimal Estimation* MIT Press, 1974

[48] A. Gómez-Pérez. *Ontological Engineering.* Tutorial, IJCAI, 1999.

[49] P. Greenspun *Scalable Systems for Online Communities* in: P. Greenspun(Ed.): Pilip and Alex's Guide to Web Publishing; Morgan Kaufman Publishers 1999

[50] G. Groh, C. Hillebrand *Location-Based Community Services* to appear

[51] G. Groh. *Applying Text-Classification Methods to the Mapping of Simple extensional Ontologies for Community Information Management* Diploma Thesis, Dept. of Computer Science, University of Kaiserslautern, March 2001

[52] G. Groh. *Ubiquitous, Wearable and Affective Computing - New Ways in Context-Sensitive Human Computer Interaction?* Seminar Thesis, Dept. of Computer Science, University of Kaiserslautern, June 1999, http://www11.in.tum.de/personen/groh/pub/seminar.pdf (URL, Aug. 2003).

[53] G. Groh, M. Koch et al. *COSMOS Planning Document* http://sunschlichter3.informatik.tu-muenchen.de/planung/planung.html (URL, Aug. 2003)

[54] G. Groh, M. Koch, N. Fremuth et al. *COSMOS Website and Project Community* http://www.cosmos-community.org (URL, Aug. 2003)

[55] G. Groh, M. Koch, C. Ehmig, R .Friess, N. Fremuth, O. Brakel, A. Tasch, et. al. *Studiosity community platform.* http://www.studiosity.de (URL, Jan. 2004).

[56] I. Gradshteyn, D. Zwillinger (Eds.) *Table of Integrals, Series and Products* Academic Press, 2000

[57] T. Gross, M. Specht *Awareness in Context-Aware Information Systems* Proc. Mensch & Computer 01, Bonn 2001.

[58] G. Görz, C-R.Rollinger, J.Schneeberger (Eds.). *Handbuch der Künstlichen Intelligenz* Oldenbourg, München, 2000.

[59] U. Hansmann, M. Nicklous, T. Stober, *Pervasive Computing Handbook* Springer, November 2000.

[60] M. Halkidi, Y. Batistakis, M. Varzirgiannis. *Cluster Validity Methods: Part I.* http://citeseer.nj.nec.com/534869.html (URL, Jan. 2003)

[61] M. Halkidi, Y. Batistakis, M. Varzirgiannis. *Cluster Validity Methods: Part II.* http://citeseer.nj.nec.com/537304.html (URL, Jan. 2003)

[62] B. Hamp, H. Feldweg *GermaNet - a Lexical-Semantic Net for German* In: Proc. of ACL workshop "Automatic Information Extraction and Building of Lexical Semantic Resources for NLP Applications", Madrid, 1997

[63] A. Hare *Handbook of Small Group Research* Free Press, New York, 1976

[64] D. Harper *Online Etymological Dictionary* http://www.etymonline.com (URL, Jul. 2003)

[65] G.Hartfiel, K. Hillmann *Wörterbuch der Soziologie* A. Kröner Verlag, Stuttgart, 1994

[66] P. Hayes (Ed.) *RDF Semantics - W3C Proposed Recommendation* http://www.w3.org/TR/2003/PR-rdf-mt-20031215/ (URL, Jan. 2004)

[67] J. Heflin (Ed.) *OWL Web Ontology Language Use Cases and Reuirements - W3C Proposed Recommendation* http://www.w3.org/TR/2003/PR-webont-req-20031215 (URL, Jan. 2004)

[68] F. Heinzle, M. Kopczynski, M. Sester *Spatial Data Interpretation for the Intelligent Access to Spatial Information in the Internet* In: 'Proceedings of 21st International Cartographic Conference', August 10-16, 2003, Durban, South Africa.

[69] W. Herkner *Lehrbuch Sozialpsychologie* Verlag H. Huber, Bern, 1991

[70] G. Hillery *Definitions of community: Areas of agreement.* Rural Sociology, 20:pp. 111–123, 1955.

[71] G. Hirst *Lexical Chains as Representations for Context for the Detection and Correction of Malapropisms* in C. Fellbaum (Ed.) - WordNet: an electronic lexical database, MIT Press, p. 305-332, 1998

[72] G. Homans *The Human Group* Reprint Transaction Pub., New York, 2001 (org. 1950)

[73] F. Höppner, F. Klawonn, R. Kruse, T. Runkler *Fuzzy Cluster Analysis* John Wiley & Sons, 1999

[74] X. Hong, M. Gerla, G. Pei, C. Chiang *A Group Mobility Model for Ad Hoc Wireless Networks* Proceedings of ACM/IEEE MSWiM'99, Seattle, WA, USA, Aug. 1999, pp.53-60.

[75] A. Hulth, J. Karlgren, A. Jonsson, H. Boström, L. Asker *Automatic Keyword Extraction Using Domain Knowledge* in A. Gelbukh (Ed.): CICLing 2001, LNCS 2004, pp. 472-482, 2001

[76] T. Ishida *Community Computing.* John Wiley and Sons, 1998.

[77] A. Jain, R. Dubes *Algorithms for Clustering Data* Prentice Hall, 1988.

[78] A. Jain, M. Murty, P. Flynn *Data Clustering: A Review* ACM Computing Surveys, Vol. 31, No. 3, Sept. 1999

[79] J. Jiang, D. Conrath *Semantic Similarity Based on Corpus Statistics and Lexcial Taxonomy* Proc. Int'l Conf. on Research in Computational Linguistics (ROCLING X) Taiwan, 1997.

[80] C. Jones et al *Spatial Information Retrieval and Geographical Ontologies: An Overview of the SPIRIT project* In: 'SIGIR 2002: Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval August 11-15, 2002, Tampere, Finland', ACM Press, pp.387 - 388.

[81] H. Kautz, B. Selman, M. Shah *Referral Web: Combining Social Networks and Collaborative Filtering.* Communications of the ACM, Vol. 40, Issue 3, pp. 63-65, March 1997

[82] M. Kifer, G. Lausen, and J. Wu. *Logical Foundations of Object-Oriented and Frame-Based Languages.* Journal of the ACM, 42:741–843, 1995.

[83] A. Kemper, A. Eickler *Datenbanksysteme.* Oldenbourg, 2001.

[84] J. Kim, D. Oard, K. Romanik *User Modeling for Information Filtering Based on implicit Feedback.* Proc. ISKO conference, Nanterre, France, 2001

[85] M. Klein, T. Furtak *Optics.* Springer 1988

[86] J. Koch *Unterstützung der Formierung und Analyse von virtuellen Communities.* PhD thesis, Dept. of Computer Science, Technische Universität München, Jul. 2002.

[87] M. Koch *Community-Unterstützungssysteme - Architektur und Interoperabilität.* Habilitation thesis, Dept. of Computer Science, Technische Universität München, February 2003.

[88] M. Koch, G. Groh, C. Hillebrand *Mobile Communities - Extending Online Communities into the Real World.* Proc. Americas Conf. on Information Systems (AMCIS2002), Dallas, TX, Aug. 2002.

[89] M. Koch, G. Groh, C. Hillebrand, N. Fremuth *Mobile Support for Lifestyle Communities.* Tech. Report Nr. 34, Lehrstuhl für allgemeine und angewandte BWL, TU-München, hrsg. v. R. Reichwald, Nov. 2002

[90] M. Koch, P. Schubert *Collaboration Platforms for Virtual Student Communities.* Proc. HICSS-36, Hawaii, Jan. 2003

[91] D. Kopjev *Implementierung eines mobilen kontextsensitiven Systems für Location Based Information Retrieval.* Project thesis (SEP), Technische Universität München, Fak. f. Informatik, LS Prof. Schlichter, 2002. Supervised by G. Groh

[92] T. Kosch, C. Schwingenschlögl, L. Ai *Information Dissemination in Multihop Inter-Vehicle Networks - Adapting the Ad-hoc On-demand Distance Vector Routing Protocol (AODV).* The IEEE 5th International Conference on Intelligent Transportation Systems Singapore, 2002

[93] T. Kosch, C. Schwingenschlögl *Geocast Enhancements for AODV in Vehicular Networks.* ACM SIGMOBILE Mobile Computing and Communications Review Juli 2002

[94] K. Kukich *Techniques for Automatically Correcting Words in Text.* ACM Computing Surveys, Vol. 24, No. 4, pp. 377-439, Dec. 1992.

[95] R. Kraut, C. Egidio, J. Galegher *Patterns of Contact and Collaboration in Scientific Research Collaboration.* in J. Galegher, R. Kraut (eds.) "Intellectual Teamwork: Social and Technological Foundations of Cooperative Work", pp. 149-172, 1990

[96] M. Lacher, G. Groh *Facilitating the Exchange of Explicit Knowledge through Ontology Mappings* Proc.of 14th International FLAIRS conference, AAAI Press, p.305-309, May 2001

[97] M. Lacher *Supporting the Exchange of Knowledge in Communities of Interest via Document Catalog Mediation* PhD thesis, Technical University of Munich, Sept. 2003

[98] M. Lamming, W. Newman *Activity-based Information Retrieval: Technology in Support of Personal Memory*, In: F.H.Vogt(ed.), Personal Computers and Intelligent Systems, Vol A-14 of IFIP 12th World Congress. Proceedings of Information Proccessing 1992, p. 68-81, 1992.

[99] M. Lamming, M. Flynn *"Forget-me-not" Intimate Computing in Support of Human Memory*, Technical Report EPC-1994-103, Rank Xerox Research Center, published in: Proceedings od FRIEND21, '94 International Symposium on Next Generation Human Interface, 2-4 Feb. 1994, Meguro Gajoen, Japan.

[100] M. Lamming, *Towards Future Personalized Information Environments*, Technical Report EPC-1994-104, Rank Xerox Research Center, published in: Proceedings od FRIEND21, '94 International Symposium on Next Generation Human Interface, 2-4 Feb. 1994, Meguro Gajoen, Japan.

[101] B. Latané, J. Liu, A. Nowak, M. Benvento, L. Zheng *Distance Matters: Physical Space and Social Impact*, Personality and Social Psychology Bulletin, 21(8), 1995, pp.795-805

[102] B. Latané *The Psychology of Social Impact*, American Psychologist, 36(4), 1981, pp. 343-356

[103] B. Latané, T. L'Herrou *Spatial Clustering in the Conformity Game: Dynamic Social Impact in Electronic Groups*, University of California, Berkeley, 2004, http://sherlock.berkeley.edu/geo_ir/PART1.html (URL, Dec. 2004)

[104] R. Larson *Geographic Information Retrieval and Spatial Browsing*, Journal of Personality and Social Psychology 70(6), pp. 1218-1230, 1996

[105] C. Leacock, M. Chodorow *Combining Local Context and WordNet Similarity for Word Sense Identification* in C. Fellbaum (Ed.) - WordNet: an electronic lexical database, MIT Press, p. 266-283, 1998

[106] T. Leckner, M. Koch, M. Lacher, R. Stegmann *Personalization meets Mass Customization - Support for the Configuration and Design of Individualized Products* Proc. 5th Intl. Conf. on Enterprise Information Systems (ICEIS), Vol. 4, pp. 259-264, France, 2003

[107] B. Liang, Z. Haas *Predictive Distance-Based Mobility Management for PCS Networks* Proc. IEEE Infocom'99, New York, 1999, p. 1377-1384

[108] H. Lieberman *Letizia: An Agent That Assists Web Browsing* Proc. 14th International Joint Conference on Artificial Intelligence (IJCAI-95), Montreal, Canada, p 924–929, 1995

[109] D. Lin *An Information-Theoretic Definition of Similarity* Proc. 15th International Conference on Machine Learning, pp. 296-304, 1998.

[110] C. Lueg, D. Fisher (Ed.) *From UseNet to CoWeb - Studying Social Information Spaces* Springer 2003

[111] B. Magnini, C. Strapparava, G. Pezzulo, A. Gliozzo *Using Domain Information for Word Sense Disambiguation* In Proc. SENSEVAL-2: Second International Workshop on Evaluating Word Sense Disambiguation Systems, pp. 111-114, Toulouse, Jul. 2001.

[112] C. Manning *Foundations of Statistical Natural Language Processing* MIT Press, Aug. 1999

[113] J. Martinez-Fernandez, A. Garcia-Serrano, P. Martinez, J. Villena *Automatic Keyword Extraction for News Finder* A. Nürnberger, M. Detyniecki (Eds.): AMR 2003, LNCS 3094, pp. 99-119, 2004

[114] A. McCallum. Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering. http://www.cs.cmu.edu/ mccallum/bow, 1996.

[115] D. McDonald *Evaluating Expertise Recommendations* Proc. 2001 Int'l ACM SIGGROUP Conf. on Supporting Group Work, Boulder, USA, pp. 214-223, 2001

[116] D. McGuiness, F. van Harmelen *OWL Web Ontology Language Overview* http://www.w3.org/TR/2003/Pr-owl-features-20031215/ (URL, Jan. 2004)

[117] G. Miller *WordNet: A Lexical Database for English* Communications of the ACM, 38(11), pp. 39-41, 1995.

[118] G. Miller, W. Charles *Contextual Correlates of Semantic Similarity* Language and Cognitive Processes, 6(1), pp. 1-28, 1991.

[119] T. Melnyk, O. Knop, W. Smith *Extremal Arrangements of Points and Unit Charges on a Sphere: Equilibrium Configurations Revisited* Canad. J. Chem. 55, 1745-1761, 1977

[120] M. Morita, Y. Shionoda *Information Filtering Based on User Behaviour Analysis and Best Match Text Retrieval* Proc. 17th Int'l ACM-SIGIR Conf. on Research and Development, pp. 272-281, Springer, NY, 1994

[121] J. Morris *Lexical Cohesion, the Thesaurus and the Structure of the Text* Tech. Report CSRI-219, Computer Systems Research Group, University of Toronto, Dec. 1988

[122] E. D. Mynatt, A. Adler, M. Ito, and V. L. ODay. *Design for network communities.* In Proc. ACM SIGCHI Conf. on Human Factors in Compt. Syst., 1997.

[123] N. N. *Das Latein Wörterbuch - Homepage* http://www.latein-woerterbuch.de (URL, Jul. 2003)

[124] N. N. *TeleAtlas - Homepage* http://www.teleatlas.com (URL, Dec. 2004)

[125] N. N. *OracleSpatial Whitepaper* http://www.oracle.com/technology/products/spatial/pdf/10_spatial_locator_twp.pdf (URL, Dec. 2004)

[126] N. N. *Homepage GermaNet* http://www.sfs.nphil.uni-tuebingen.de/lsd (URL, Jan. 2003)

[127] N. N. *Friedscout24 Community Platform* http://www.friendscout24.de/lsd (URL, Jan. 2004)

[128] N. N., *Cassiopeia Website.* www.cassiopeia.com (URL, Apr. 2003)

[129] N. N. *GSM Guide* http://www.umtslink.at/GSM-Start.htm (URL, Sep. 2003).

[130] N. N. *Wiki Definition in Wikipedia.* http://www.wikipedia.org/wiki/WikiWiki (URL, Aug. 2003).

[131] N. N. *jetzt.de community platform.* http://www.jetzt.de (URL, Jan. 2004).

[132] N. N. *The Network Simulator - ns-2.* http://www.isi.edu/nsnam/ns/ (URL, Nov. 2003).

[133] N. N. *Amazon Store.* http://www.amazon.com (URL, Dec. 2003).

[134] N. N. *Yahoo Personals Dating Community platform.* http://personals.yahoo.com (URL, Jan. 2004).

[135] N. N. *MTV Community platform.* http://www.mtv.com/community/ (URL, Jan. 2004).

[136] N. N. *MyScene Community platform.* http://myscene.urbanrave.com (URL, Jan. 2004).

[137] N. N. *Microsoft Encarta Dictionary.* http://http://encarta.msn.com/encnet/features/dictionary/dictionaryhome.aspx (URL, Feb. 2004).

[138] N. N. *Open Survey Pilot.* http://surveypilot.de (URL, Jan. 2004).

[139] N. N. *Microsoft Netscan.* http://netscan.research.microsoft.com (URL, Mar. 2004).

[140] N. N. *Oroinc NNTP Implementation in Java (Freeware).* http://www.oroinc.com (URL, Nov. 2004).

[141] N. N. *BSCW shared workspace system home page.* http://bscw.gmd.de (URL, Apr. 2004).

[142] N. N. *CVS collaborative versioning tool home page.* http://www.cvshome.org (URL, Apr. 2004).

[143] N. N. *AOL Instant Messenger home page.* http://www.aim.com (URL, Apr. 2004).

[144] G. Navarro *A Guided Tour To Approximate String Matching* ACM Computing Surveys, Vol.33, No. 1, March 2001, pp.31-88.

[145] G. Nahemow, M. Lawton *Similarity and Propinquity in Friendship Formation* Journal of Personality and Social Psychology, 32(2), 1975, pp.205-213

[146] R. Neches, R. Fikes, T. Finin, T. Gruber, et al. *Enabling Technology for Knowledge Sharing.* AI Magazine, 12(3):36–56, 1991.

[147] H. Ney *Have we found the holy grail?.* MT Summit IX, New Orleans, Panel Discussion, Slides, Sep. 2003 http://www.amtaweb.org/summit/MTSummit/FinalPapers/MT_Summit_Panel_18Sep03.pdf (URL Feb. 2004)

[148] D. Nichols *Implicit Rating and Filtering* Proc. 5th DELOS Workshop on Filtering and Collaborative Filtering, pp. 31-36, Sophia Antipolis, France, 1997

[149] H. Ogatha *Computer Supported Social Networking for Augmenting Cooperation* Computer Supported Coorperative Work 10, 2001, pp. 189-209, Kluwer Academic Publishers.

[150] Open GIS Consortium (OGC) *Geography Markup Language (GML) - Committee Draft* http://opengis.net/gml/ (URL, Dec. 2004)

[151] G. Olsen, J.Olsen *Distance Matters* Human Computer Interaction, 15(2/3), 2000, pp. 39-178

[152] F. Piller *Customer interaction and digitizability - a structural approach to mass customization* in: Rautenstrauch et al. (ed.): Moving towards mass customization, Springer: Heidelberg/Berlin/New York, 2002.

[153] S. Pradhan, C. Brignone, J. Cui, A. McReynolds, M. Smith *Websigns: Hyperlinking Physical Locations to the Web* IEEE Computer, 34(8), pp. 42-48, Aug. 2001

[154] R. Reichwald, N. Fremuth, M. Ney *Mobile Communities.* in: R. Reichwald (Ed.) Mobile Kommunikation, Gabler, Wiesbaden 2002.

[155] R. Reichwald, N. Fremuth, M. Ney *Kundenintegrierte Entwicklung mobiler Dienste.* in: R. Reichwald (Ed.) Mobile Kommunikation, Gabler, Wiesbaden 2002.

[156] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, J. Riedl *GroupLens: An Open Architecture for Collaborative Filtering of Netnews.* in: Proc. ACM 1994, Conference on CSCW, pp. 175-186, New York, 1994

[157] P. Resnick *Using Information Content to Evaluate Semantic Similarity.* in: Proc. 14th Joint Int'l Conf. on Artificial Intelligence, pp. 448-453, Montreal, Aug. 1995

[158] H. Rheingold. *The Virtual Community.* Addison-Wesley, 1993.

[159] M. Richter *Prinzipien der künstlichen Intelligenz.* Teubner, Frankfurt, 1992

[160] B. Rhodes. *The Wearable Remembrance Agent - A System for Augmented Memory.* Proc. 1st Int'l Symposium on Wearable Computers (ISWC '97), Cambridge MA, Oct. 1997, p. 123-128.

[161] T. Runkler. *Information Mining.* Vieweg, 2000.

[162] T. Runkler J. Bezdek *Alternating Cluster Estimation: A New Tool for Clustering and Function Approximation.* IEEE Transactions on Fuzzy Systems, Vol. 7 Nr. 4 Aug 1999, pp. 377-393

[163] T. Runkler J. Bezdek *RACE: Relational Alternating Cluster Estimation and the Wedding Table problem* in: W. Brauer (ed.) Fuzzy-Neuro-Systems 98, Bd. 7 of "Proceedings in Artificial Intelligence", pp. 330-337, Munich, Mar 1998

[164] W. Sack *Conversation Map: A Content-Based Usenet Newsgroup Browser* Proc. ACM Conf. on Intelligent User Interfaces (IUI 00), pp. 233-240, 2000

[165] G. Salton, C. Buckley *Stopword List* http://www.lextek.com/manuals/onix/ stopwords2.html (URL, Feb. 2004)

[166] B. Schilit, N. Adams, R. Want *Context-Aware Computing Applications* Proc. of the IEEE Workshop on Mobile Computing Systems and Applications, Santa Cruz, CA, Dec. 1994, p 85-90.

[167] A. Schmidt *Implicit Human Computer Interaction Through Context* Personal Technologies, Vol 4(2), June 2000

[168] P. Schubert, *Virtuelle Transaktionsgemeinschaften im Electronic Commerce.* Josef Eul Verlag, Lohmar, 2000

[169] D. Schuler *New Community Networks - Wired for Change.* Addison Wesley, NY, 1996

[170] N. Sloane. *Spherical Codes.* http://www.research.att.com/ njas/packings/index.html (URL, Mar. 2003)

[171] P. Smart, A. Abdelmoty, C. Jones *An Evaluation of Geo-Ontology Representation Languages for Supporting Web Retrieval of Geographic Information.* In: 'Proceedings of the GIS Research UK 12th Annual Conference', Norwich, UK, pp. 175-178.

[172] M. Smith, A. Fiore *Visualization Components for Persistent Conversations.* Proc. CHI 2001, pp. 136-143, Mar. 2001

[173] K. Sparber, G. Groh *Datamining in Newsgroups for the Modelling of Binary and N-Ary Relations between Users.* Systementwicklungsprojekt, Fakultät für Informatik, LS Prof. Schlichter, TU-München, Oct. 2003

[174] S. Staab. *Intelligent Systems on the World Wide Web.* Lecture Slides, University of Karlsruhe, 2000.

[175] C. Stegbauer (Sociologist, Univ. of Frankfurt, Germany) *(Personal communication).* GOR04 Conference, Duisburg, March 2004.

[176] R. Stegmann *Natural Language Dialogs for User Profile Acquisition (Working Title).* Dissertation, TU München Fak. f. Informatik, 2005 (to appear)

[177] A. Steger, T. Schickinger *Diskrete Strukturen 2.* Springer, 2001

[178] A. Stephan *Emergenz: Von der Unvorhersagbarkeit zur Selbstorganisation.* Dresden Univ. Press, 1999

[179] R. Studer, T. Benjamins, and D. Fensel. *Knowledge Engineering: Principles and Methods.* Data and knowledge Engineering, (25):161–197, 1998.

[180] B. Swartout, R. Patil, K. Knight, and T. Russ. *Toward distributed Use of Large-Scale Ontologies.* In Ontological Engineering, AAAI-97 Spring Symposium Series, pages 138–148, 1997.

[181] H. Schlichter, U. Baumgarten, O. Brakel, M. Daum, G. Groh, H. Hillebrand, A. Tasch, J. Leimeister, H. Krcmar, R. Reichwald *Abschlussbericht COSMOS* (to appear)

[182] H. Tajfel and J. Turner *The social identity theory of inter-group behavior* In S. Worchel and L. W. Austin (eds.), Psychology of Intergroup Relatio ns. Chigago: Nelson-Hall, 1986

[183] F. Tönnies *Gemeinschaft und Gesellschaft, Grundbegriffe der reinen Soziologie.* Wissenschaftl. Buchgesellschaft Darmstadt, 1991; (Originalwerk von 1887).

[184] J. Turner *Towards a cognitive redefinition of the social group* . In H. Tajfel (ed.), Social Identity and Intergroup Relations. Cambridge: Cambridge University Press, 1982

[185] W. van Vliet, J.Burgers, *Community in Transition: From the Industrial to the Postindustrial Era* in I. Altman, A. Wandersman (eds.): Neighbourhood and Community Environments, Plenum Press, NY, 1987

[186] M. Uschold and M. Gruninger. *Ontologies, Principles, Methods and Applications.* Knowledge Engineering Review, 11(2):93–155, 1996.

[187] J. Walter *Interpersonal Effects in Computer Mediated Communication: A Relational Perspective* Communication Research 19 (1992), p. 52-90

[188] S. Wassermann, K. Faust. *Social Network Analysis. Methods and Applications.* Cambridge University Press, 1994.

[189] P. Watzlawick, J. Beavin, D. Jackson, *Menschliche Kommunikation. Formen, Störungen, Paradoxien.* Huber (Bern,Stuttgart,Toronto) 1990

[190] M. Weiser *The Computer for the 21st Century.* Scientific American, p. 94-104, Sept. 1991

[191] C. Welty. The Ontological Nature of Subject Taxonomies. In *Proc. International Conference on Formal Ontology in Information Systems (FOIS)*, 1998.

[192] J.-R. Wen, J.-Y. Nie, H.-J. Zhang *Query Clustering Using User Logs.* ACM Transactions on Information Systems, Vol. 20, No. 1, Jan. 2002, pp. 59-81

[193] L. Whyte. *Unique Arrangement of Points on a Sphere.* Amer. Math. Monthly 59, 606-611, 1952

[194] W. Woerndl *Privatheit bei dezentraler Verwaltung von Benutzerprofilen.* PhD thesis, Technische Universität München, Falultät für Informatik, Aug. 2003

[195] E. Wong, B. Hajek *Stochastic Processes in Engineering Systems.* Springer, 1985

[196] C. Womser-Hacker *Einführung in die Informationswissenschaft.* Vorlesung 2003, Uni Hildesheim, http://www.uni-hildesheim.de/ einf_iw/folien/S14_Multimedia.PDF (URL, Apr. 2004)

[197] Z. Wu, M. Palmer *Verb Semantics and Lexical Selection.* Proc. 32nd Ann. Meeting of the Association of Computational Linguistics, pp 133-138, Las Cruces, Jun. 1994

[198] R. Zajonc *Attitudinal Effects of Mere Exposure.* Journal of Personality and Social Psychology, Monograph Supplement 9(2, part 2) pp. 1-27, 1968

[199] X. Zeng, R. Bagrodia, M. Gerla *GloMoSim: a Library for Parallel Simulation of Large-scale Wireless Networks.* Proceedings of the 12th Workshop on Parallel and Distributed Simulations – PADS '98, May 26-29, 1998, Banff, Alberta, Canada. See also http://pcl.cs.ucla.edu/projects/glomosim/ (URL, Nov. 2003)

[200] H.-J. Zimmermann *Fuzzy Set Theory - and Its Applications.* Kluwer Academic Publishers, 1991

Abbreviations

- **CIKS**: Collaborative Information- or Knowledge Space ..... *Set of Knowledge- or Information-Items in a Community (see 1.2.2, 1.2.3, 1.3.2)*
- **CMC**: Computer Mediated Communication ..... *communication (mostly textual) with the help of a computer (see 1.2.2)*
- **CSCW**: Computer Supported Collaborative Work ..... *Form of collaboration and Computer-Science discipline*
- **CVS**: Concurrent Versioning System ..... *Versioning Tool*
- **FCM**: Fuzzy C-Means ..... *Fuzzy clustering algorithm (see 5.2.1.1)*
- **GML**: Geographic Markup Language ..... *Declarative Language for Spatial Relations*
- **FOL**: First Order Logic ..... *First Order Predicate Logic*
- **GAR**: Generalized Abstraction Relation ..... *Union of special types of abstraction relations (e.g. in WordNet)*
- **GPS**: Global Positioning System ..... *Satellite based positioning system*
- **GPRS**: General Packet Radio Service ..... *Packet-oriented mobile data transmission standard*
- **GSM**: Groupe Spécial Mobile ..... *European mobile telecommunication standard developed by the Groupe Spécial Mobile* working group of the Conference of European Posts and Telegraphs (CEPT)
- **MIR**: Multiple Instance Removal ..... *Converting a Multi-Set into a Set*
- **MMC**: Mixed-Real-World-Virtual-Mobile Community ..... *Community that communicates face-to-face and also with all available types of media (see 2.3)*
- **MUD**: Multi-User Dungeon ..... *Gaming and Communication Environment on the Web*
- **MVV**: Münchner Verkehrs Verbund ..... *Munich public transportation company*
- **MST**: Minimum Spanning Tree Clustering ..... *Crisp clustering algorithm (see 5.1.2.3)*
- **MTV**: Music Television ..... *Music Television company and TV channel*
- **NLP**: Natural Language Processing ..... *Discipline of processing language with computers*

- **OWL**: Ontology Web Language ..... *Description-Logics-based declarative ontology language*
- **PC**: Partition Coefficient ..... *Fuzzy cluster validation measure (see 5.2.1.5)*
- **PDA**: Personal Digital Assistant ..... *Small mobile computer*
- **PE**: Partition Entropy ..... *Fuzzy cluster validation measure (see 5.2.1.5)*
- **POI**: Point Of Interest ..... *Entity associated with a location in a digital map*
- **RACE**: Relational Alternating Cluster Estimation ..... *Fuzzy relational clustering algorithm (see 5.2.1.4)*
- **RACE-SA**: Relational Alternating Cluster Estimation with Simulated Annealing ..... *Fuzzy relational clustering algorithm with special optimization (see 5.2.2.1)*
- **RBG**: Red, Blue, Green ..... *Color Model*
- **RDF(S)**: Resource Description Framework (Schema) ..... *Declarative Semantic-Web Ontology language for semantic nets*
- **RFAO**: Relational Fuzzy Alternating Optimization ..... *Fuzzy Clustering Algorithm (see 5.2.2.2)*
- **RWC**: Real world communication ..... *Communicaton that does not necessarily involve computers (see 1.2.2)*
- **SAHN**: Sequential Agglomerative Hierarchical Non-Overlapping Clustering ..... *Class of crisp clustering algorithms (see 2.1.2.3)*
- **SCVS**: Socially Motivated Cluster Validation and Selection ..... *Cluster validation and selection algorithm with social heuristics (see 5.1.3.3)*
- **SUMI**: Simple Urban Mobility Simulator..... *Mobility Model for Groups and Individuals in an Urban Environment (see 3)*
- **SQMSE**: Square-Root of the Mean Square Error ..... *Statistical Measure (see 4.7.7.1)*
- **SR**: Stop-Word Removal ..... *Removing "and", "or", "me" etc.*
- **TF/IDF**: Term Frequency / Inverse Document Frequency ..... *Term weighting scheme in Information Retrieval (see 4.6.2, 4.8.3.1)*
- **UMTS**: Universal Mobile Telecommunications System ..... *Mobile telecommunication standard*
- **URI**: Uniform Ressource Identifier ..... *Global identifier in mark-up languages and rsp. declarative standard*
- **WML**: Wireless Markup Language ..... *Markup language for WAP pages*
- **WSD**: Word Sense Disambiguation ..... *Techniques for differentiating senses of a word with a computer (see 4.7.4)*
- **XSLT**: Extensible Stylesheet Language - Transformations ..... *XML language for the transformation of XML-trees*

List of Tables

3.1 Transition probabilities for the mobility states 69
3.2 Average stay time D_i^{estimate} in the 6 Mobility states and standard deviations . . . 70
3.3 Average speed μ_i and estimated standard deviation σ_i in the mobility states i . . 72

4.1 Spelling errors in the interest test collections 101
4.2 The Average Ratings and Standard Deviations of Survey 1, 2 and 3 for all participants in the survey 120
4.3 Varying the settings for α and β 121
4.4 Varying the settings for α and β 122

5.1 Varying the Clustering Algorithm 146
5.2 Varying the Distance Measure 147
5.3 Varying the SCVS parameters τ_l and τ_v 149
5.4 Comparison between period-modulated and pure membership-based quality measures of the abstract group detection algorithm 160
5.5 Overall Members, periods and average Score of the Abstract Actual and Found Groups resulting from the procedure shown in figure 5.10 161
5.6 Overall Members, periods and average Score of the Abstract Actual and Found Groups resulting from the procedure shown in figure 5.10 162
5.7 Comparing the alternatives for the similarity measure for comparing Ad-Hoc-Groups with respect to their members and the two alternatives for period detection 163
5.8 Running Fuzzy C-Means and RACE on the test data 174
5.9 RACE and Drawing from samples 174
5.10 Performance of RACE-SA on the test data 177
5.11 Performance of PC and PE of 20 runs with RACE-SA and RFAO on the four cluster test data 178
5.12 Results of Group Detection on 20×20 Part of the Survey Collection 181
5.13 Results of Group Detection on 20×20 Part of the Dating Collection 182
5.14 Test Corpuses for Communication Media with Tree-Like Structure 184

C.1 The elements from the two collections "Survey Collection" and "Dating Collection" and the list of free text interest phrases (figure C.5) used in the surveys 1, 2 and 3 219
C.2 The Results of Survey 1 228
C.3 The Results of Survey 2 229
C.4 The Results of Survey 3 230
C.5 The Average Results and Standard Deviations of Survey 1, 2 and 3 (Part I) . . . 231
C.6 The Average Results and Standard Deviations of Survey 1, 2 and 3 (Part II) . . 232
C.7 The Results of the similarity measures corresponding to Survey 1, 2 and 3 (Part I) 233
C.8 The Results of the similarity measures corresponding to Survey 1, 2 and 3 (Part II) 234

List of Figures

1.1 Some examples of knowledge and information in various degrees of explicitness and formalization . . . 12
1.2 Qualitative views on information flows to and from the web . . . 17
1.3 Using GSM cells for localization . . . 20
1.4 COSMOS Studiosity platform software architecture . . . 25
1.5 COSMOS Studiosity platform screenshots, Part (I) . . . 28
1.6 COSMOS Studiosity platform screenshots, Part (II) . . . 29
1.7 COSMOS Studiosity mobile platform screenshots, Part (I) . . . 30
1.8 COSMOS Studiosity mobile platform screenshots, Part (II) . . . 31

2.1 Social geometries described in [103] . . . 40
2.2 Cliques, n-cliques, n-clans and n-clubs in undirected graphs . . . 43
2.3 Algorithm for single-link and complete-link sequential agglomerative hierarchical non-overlapping clustering . . . 45
2.4 Threshold graphs from the graph characterized by weight-matrix A in (2.6) . . . 47
2.5 Dendrograms resulting from applying the algorithm of figure 2.3 to the threshold graphs from figure 2.5 . . . 47

3.1 Advanced Positioning through additional triangulation calculations & Event-based, proactive transfer of the localization information to the community server 60
3.2 Basic Continuous Mobility Models . . . 63
3.3 Two trajectories resulting from the Gauss-Markov Mobility Model computed with SUMI . . . 67
3.4 Boundary of the simulation area and corresponding changes in the μ_θ value . . . 67
3.5 Markov Chain for Mobility State Selection . . . 68
3.6 Coarse estimation of average velocity of public transport systems . . . 71
3.7 Markov Chain for Resting and Moving . . . 72
3.8 The structure of a periodic group schedule . . . 77
3.9 The social neighborhood of a node (person) . . . 78
3.10 Point reach tolerance and oscillations . . . 80

3.11 SUMI simulation with 5 nodes . . . 82
3.12 SUMI simulation with 5 nodes . . . 83
3.13 Inconsistencies through discretization . . . 84

4.1 Example for a List Of Choices . . . 89
4.2 Collection of Free-text Interest Phrases . . . 92
4.3 Two possible simple subject taxonomies for the interest choices shown in the right part of figure 4.1 . . . 94
4.4 Two further interest area ontologies that make use of "is-a" relations, "part-of" relations and "instance-of" relations . . . 95
4.5 Algorithm for a Similarity Measure $\mathrm{sim}(x^{(j_1)}, x^{(j_2)})$ between List-of-Choice Vectors in a Topic Taxonomy with Generalized Abstraction Relations . . . 99
4.6 An example for the first part of the algorithm of figure 4.5 . . . 100
4.7 WordNet . . . 107
4.8 Path lengths between Synsets, their nearest common generalized super-concept and the root node . . . 113
4.9 Part I of the Algorithm for a Similarity Measure for Sets of Free Text Interest Phrases $\mathrm{sim}(\mathcal{X}^{(j_1)}, \mathcal{X}^{(j_2)})$ . . . 117
4.10 Part II of the Algorithm for a Similarity Measure for Sets of Free Text Interest Phrases $\mathrm{sim}(\mathcal{X}^{(j_1)}, \mathcal{X}^{(j_2)})$ . . . 118
4.11 Transformation to Logical View . . . 126
4.12 Graph computed from the reply relation . . . 128
4.13 The function $1 - 1/(1 + m_{k_1 \to k_2} m_{k_2 \to k_1} \exp(-\frac{|m_{k_1 \to k_2} - m_{k_2 \to k_1}|^2}{\sigma^2}))^q$ . . . 129

5.1 Algorithm for K-Means Clustering . . . 138
5.2 Simple Visualization of a SUMI Simulation Step . . . 140
5.3 Simple Visualization of a SUMI Simulation Step with SCVS Ad-Hoc-Group detection . . . 146
5.4 Varying the SCVS parameters τ_v and τ_l . . . 150
5.5 Varying the SCVS parameters τ_v and τ_l . . . 151
5.6 The occurrences of two Ad-Hoc-Groups over several subsequent runs of the group detection procedure . . . 153
5.7 The function f_b and its Fourier transform $\mathscr{F} f_b$ . . . 154
5.8 The function f_Σ and the absolute of its Fourier transform $|\mathscr{F} f_\Sigma|$ . . . 155
5.9 The function $\gamma(\nu) = |\frac{\sin(N\pi\nu a)}{\sin \pi\nu a}|$ . . . 156
5.10 Algorithm for Computing Abstract Groups . . . 159
5.11 Algorithm for Fuzzy C-Means Clustering . . . 167
5.12 Simple Visualization of a SUMI Simulation Step with Fuzzy C-Means Ad-Hoc-Group detection . . . 168

5.13 RACE part I 170
5.14 RACE part II 171
5.15 Creating absolute and relative input patterns for the Fuzzy Clustering algorithms with the location setter 173
5.16 Determining the new prototype index from a random number 176
5.17 The first 20 free text interest phrases sets from the "Survey Collection" 179
5.18 The first 20 List-Of-Choice interest vectors from the "Dating Collection" 181

6.1 Visualizing groups 189
6.2 Visualizing Groups (II) 190
6.3 Visualizing Groups (III) 190
6.4 Visualizing fuzzy groups with the help of color encoding 192
6.5 Input and retrieval of an information item with location relevance 201

C.1 The first 50 free text interest phrases sets from the "Survey Collection" 220
C.2 Free text interest phrases sets 51-100 from the "Survey Collection" 221
C.3 The first 50 List-of-Choice interest vectors from the "Dating Collection" 222
C.4 List-of-Choice interest vectors 51-100 from the "Dating Collection" 223
C.5 The free text interest phrases used for survey 3 224
C.6 The first two pages of the questionnaire of survey 1 225
C.7 The first two pages of the questionnaire of survey 2 226
C.8 The first page and the third page of the questionnaire of survey 3 227

Danke

Mein besonderer Dank gilt all denen, die mich in irgendeiner Weise beim Zustandekommen dieser Arbeit unterstützt haben. Insbesondere meinen lieben Hiwis Cand.Inform. Rene Friess, Cand.Inform. Christian Ehmig und Dipl.Inform. Markus Geiger, meinen liebsten Freunden Dipl.Inform. Cand.Psych. Cand.Phil. Cand.Math. Cand.Phys. Schachlieh Tone, Dr.rer.nat. Dipl.-Lebchem. MDRA Miriam Gensler, Dr.rer.nat. Dipl.Inform. MSc Martin Lacher, der entzückenden Dipl.Kffr.(BA) Dipl.Komm.-Des. Julia Kiehlneker, meinen lieben Kollegen Dipl.Inform. Peter Breitling, Dr.rer.nat. Dipl.Inform. Wolfgang Wörndl, MA.Comp.Ling. Rosy Stegmann, Dipl.Inform. Frank Schuetz, Dr.rer.nat. Dipl.Math. Michael Galla, Dipl.Inform. Elena Paslaru, Dipl.Oec. Miriam Daum, Dr.-Ing. Andreas Donaubauer, Dr.rer.nat. Dipl.Inform. Jürgen Koch, Dipl.-Ing. Andreas Matheus, Dipl.Inform. Weilun Zhuang, Friedel Bunke, Evelyn Gemkow, PD Dr.rer.nat.habil. Dipl.Inform. Michael Koch, Dipl.Soz.-Wi. Andreas Tasch, Dipl.Soz.-Wi. Olli Brakel und Dipl.Inform. Thomas Schöpf.

Alles was ich in meinen fachlichen Gebieten weiss (und vor allem auch das Wissen darüber, was ich nicht weiss) haben mir meine lieben Professoren vermittelt. Hier sind besonders hervorzuheben: Herr Prof.Dr.rer.nat. V.F.Müller, Herr Prof.Dr.rer.nat. W.Rühl, Herr Prof.Dr.rer.nat. J.Korsch und Herr Prof.Dr.rer.nat. O.Mayer.

Ebenfalls bedanken möchte ich mich bei den Herren Prof.Dr.rer.nat. K.Madlener und Prof. Dr.rer.nat. J.Nehmer (beide Universität Kaiserslautern) für die Möglichkeit, meine Diplomarbeit in Informatik in München machen zu können.

In besonderer Weise gilt mein Dank auch Herrn Prof.Dr.rer.nat. Uwe Baumgarten und in ganz ganz besonderer und ganz herzlicher Weise natürlich auch meinem Doktorvater Herrn Prof.Dr.rer.nat. Johann Schlichter, der mir eine ganz wunderbare Zeit an seinem Lehrstuhl ermöglicht hat.

Am meisten haben aber natürlich meine tadellosen, lieben Eltern Helga Groh und Johannes Groh beigetragen. Ihnen gebührt selbstverständlich und in bester Tradition der allerschönste Dank.