# Quantifying and Protecting Location Privacy

THÈSE N$^{O}$ 5622 (2013)

PRÉSENTÉE LE 8 MARS 2013

À LA FACULTÉ INFORMATIQUE ET COMMUNICATIONS
LABORATOIRE POUR LES COMMUNICATIONS INFORMATIQUES ET LEURS APPLICATIONS 1
PROGRAMME DOCTORAL EN INFORMATIQUE, COMMUNICATIONS ET INFORMATION

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

Reza SHOKRI

acceptée sur proposition du jury:

Prof. M. Grossglauser, président du jury
Prof. J.-P. Hubaux, directeur de thèse
Dr G. Danezis, rapporteur
Prof. J.-Y. Le Boudec, rapporteur
Prof. V. Shmatikov, rapporteur

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Suisse
2013

# Preface

It is well known that many technologies have downsides that are initially overlooked or underestimated: engines and heating systems lead to exhaustion of fossil resources and climate change, road traffic kills people by the thousands, etc. Information technology is no exception as it has notably paved the way to an unprecedented assault on privacy. Privacy was already identified as a major issue as early as the last decade of the nineteenth century, with the emergence of photography. Today the pervasiveness of digital systems has brought the concern to a completely different level. In particular, the total number of cellular phones now in operation already exceeds six billion, with a growing proportion being smart phones. This means that virtually everyone can be tracked by their cellular operator and by a growing number of location-based service (LBS) providers.

This thesis revolves around the crucial topic of location privacy. It presents an analytical framework for the location privacy of LBS users. In such a setting, users share their location data and complementary application-dependent information with an LBS provider. However, users should be concerned about possible third-party observers who track their shared locations and therefore violate their privacy. This work relies on privacy enhancing technologies that do not require changing the architecture of LBSs, which I consider to be the most reasonable assumption. This means that the considered protection mechanisms (activated by the end users) are based on data perturbation. This perturbation can be achieved by either blurring the reported location data or swapping the identifiers (pseudonyms) of end users. This thesis defines users' mobility models, their access patterns to location-based services, and their privacy and service-quality requirements. Based on Bayesian techniques and on Markov modeling, it provides methodologies for quantifying and protecting the location privacy of mobile users.

This work is particularly remarkable because it is the first convincing contribution that quantifies location privacy and because it contains a very elegant game-theory based model that makes it possible for the service provider and the end user to compute their strategies in order to optimize their (conflicting) utilities. Another outstanding achievement is the open-source tool (the "Location Privacy Meter") that embodies the designed algorithms.

The papers underpinning this work have been published in the best conferences in the field (including IEEE S&P and ACM CCS) and one paper was the runner-up for the Award for Outstanding Research in Privacy Enhancing Technologies (PET Award 2012). Furthermore, the PhD committee nominated this thesis for the School of Computer and Communication Sciences' award for the best PhD thesis (the Patrick Denantes Prize).

This thesis is a must read for anyone doing research in location privacy.

Jean-Pierre Hubaux, January 2013

# Abstract

Recent developments in information and communication technologies have been profound and life-changing. Most people are now equipped with smart phones with high computation power and communication capabilities. These devices can efficiently run multiple software applications in parallel, store a non-negligible amount of (personal) user data, process various sophisticated sensors and actuators, and communicate over multiple wireless media. Furthermore, they are commonly equipped with high-precision localization capabilities based, for example, on a GPS receiver or on triangulation with nearby base stations or access points. Mobile applications take advantage of this feature to provide location-based services to users.

The ever-increasing usage of these personal communication devices and mobile applications, although providing convenience to their owners, comes at a very high cost to their *privacy*. Interacting with location-based services (LBSs) leaves an almost indelible digital trace of users' whereabouts. Moreover, the contextual information attached to these traces can reveal users' personal habits, interests, activities, and relationships. Consequently, exposure of this private information to third parties (such as service providers) escalates their power on individuals, and opens the door to various misuses of users' personal data.

Individuals have the right, and should also have the means to control the amount of their private (location) information that is disclosed to others. In the context of location-based services, various privacy enhancing mechanisms, such as location obfuscation and user anonymization, are proposed in the literature. However, the existing *design* methodologies for location-privacy preserving mechanisms do not consistently model users' (privacy and service quality) requirements together with the adversary's knowledge and objectives. Protection mechanisms are instead designed in an ad hoc manner and irrespective of the adversary model. Consequently, there is a mismatch between the goals and results of these protection mechanisms. Furthermore, the *evaluation* of privacy preserving mechanisms and their comparison remain problematic because of the absence of a systematic method to quantify them. In particular, the assumptions about the adversary model tend to be incomplete, with the risk of a possibly wrong estimation of the users' location privacy. Arguably, the lack of a generic analytical framework for specifying protection mechanisms and for evaluating location privacy is evident. The absence of such a framework makes the design of effective protection mechanisms and the objective comparisons between them impossible.

In this thesis, we address these issues and provide solutions for a systematic quantification and protection of location privacy. To this end, we construct an analytic framework for location privacy. We formalize users' mobility model, their access pattern to location-based services, and their privacy and service quality requirements. We also model location-privacy preserving mechanisms as probabilistic functions that obfuscate users' (location and identity) information before being shared with location-based services. Moreover, in order to quantify users' location privacy, we propose inference mechanisms that measure users' information

leakage to third parties. They combine various pieces of information about users and estimate (by establishing a probabilistic belief on) users' private information (e.g., their location at a given time). Therefore, we propose the adversary's expected estimation error as, arguably, the right *metric* for location privacy.

In our inference framework, we formalize the adversary's *prior knowledge* on users, his *observation* (on the users' accesses to LBS), and his inference *objectives* (e.g., re-identifying or localizing users). We assume that adversary constructs a (mobility) profile for each user, to be used in his inference attacks. We make use of statistical tools to construct these profiles, given users' partial traces. Moreover, we model the *inference attacks* as the estimation of users' actual locations, given their profiles and their LBS accesses (observed by the adversary). We mainly use Bayesian inference to perform the estimation. In particular, we use known inference algorithms for hidden Markov models to design de-anonymization, localization, and tracking attacks. To cover more adversary's objectives, we propose an algorithm for generic location inference attacks, based on Markov-chain Monte-Carlo methods.

We also provide a software tool: the *Location-Privacy and Mobility Meter* (LPM). It is designed based on our formal framework for evaluating the effectiveness of various location-privacy preserving mechanisms and quantifying users' location privacy. As an example, using LPM, we validate the efficacy of existing location obfuscation and anonymization mechanisms on real location traces. We show that users' location privacy measured by existing popular metrics, k-anonymity and entropy, is not correlated with the adversary's success (in learning users' private information), thus these metrics are inappropriate as privacy metrics. Our results also confirm that anonymization alone is a weak location-privacy preserving mechanism. Moreover, our results show how the resilience of a protection mechanism varies with respect to different inference attacks. Hence, it is a necessity for privacy protection mechanisms to be designed with concrete attack objectives in mind.

Relying on these findings, we design optimal location obfuscation techniques tailored against localization attacks. A user needs a *protection mechanism* that maximizes her location privacy. This is at odds with the objectives of the adversary who designs inference attacks that minimize his estimation error. We propose a game-theoretic methodology that models the conflicting objectives of user and adversary simultaneously. More precisely, we model the problem as a Bayesian Stackelberg game and solve it by using linear programming. In the optimization problem, users constrain the protection mechanism to respect their service quality requirements. This enables us to find the optimal point in the tradeoff curve between privacy and service quality that satisfies both user privacy and service quality requirements. Our results indicate that anticipating for the inference attacks and considering the adversary's knowledge lead to the design of more effective protection mechanisms.

This thesis is a step towards a more systematic modeling, analysis, and design of (location) privacy enhancing technologies. We believe that our analytical approach can be used to quantify and protect privacy in scenarios and domains that are not covered in this thesis.

**Keywords:** location privacy, mobile networks, location-based services, inference attacks, location-privacy preserving mechanisms, privacy metric, Bayesian inference, hidden Markov models, game theory, Bayesian Stackelberg game

# Résumé

Le développement récent des technologies de l'information et de la communication a révolutionné notre vie quotidienne. La plupart des individus sont désormais munis de téléphones mobiles avec des capacités de calcul et de communication élevées. Ces appareils peuvent faire fonctionner de nombreuses applications en parallèle, stocker une quantité non-négligeable de données personnelles, traiter des données de senseurs, et communiquer via différents canaux sans-fil. En outre, ils sont généralement équipés de moyens de localisation très précis basés, par exemple, sur le récepteur GPS, ou sur la triangulation avec des points d'accès Internet ou des stations de base. Les applications mobiles ont mis à profit cette nouvelle fonction en fournissant aux utilisateurs mobiles de nouveaux services basés sur la localisation.

L'explosion de l'utilisation de ces téléphones et applications mobiles, bien qu'apportant une certaine commodité à leurs utilisateurs, a aussi mis sérieusement en danger la vie privée. Nos interactions avec les services de localisation laissent une trace numérique quasi indélébile de nos déplacements. L'information contextuelle liée à ces traces peut révéler aux fournisseurs de services les habitudes, intérêts, activités et relations des utilisateurs. L'exposition de ces données privées à des tierces parties aggrave leur pouvoir sur les individus, et ouvre la voie à de nombreux abus. Les individus ont le droit, et devraient surtout avoir les moyens, de contrôler la quantité d'information (de localisation) privée qui est divulguée aux autres. Différents mécanismes de protection des données privées dans les services de localisation (SL) ont été proposés, tels que l'embrouillement de la localisation et l'anonymisation des utilisateurs. Cependant, l'absence d'un cadre analytique générique pour spécifier les mécanismes de protection et pour évaluer la protection des données de localisation est manifeste. Le manque d'un tel cadre ne permet pas la comparaison objective entre les différents mécanismes de protection. De plus, les méthodologies existantes de conception de mécanismes de protection des données de localisation ne prennent pas en considération les besoins (de qualité de service et de protection) des utilisateurs en même temps que les connaissances et objectifs de l'adversaire. Nous proposons d'examiner ces problèmes dans cette thèse.

Premièrement, nous développons un cadre, un modèle, unifié pour la protection des données de localisation. En ce qui concerne les utilisateurs, nous formalisons leur modèle de mobilité, leur façon d'accéder aux services de localisation, ainsi que leurs besoins de qualité de service et de protection de leurs données privées. Nous formalisons également les mécanismes de protection des données de localisation comme des fonctions probabilistes qui brouillent les informations de localisation et l'identité des utilisateurs avant de les partager avec des services de localisation. En ce qui concerne l'adversaire, nous formalisons ses connaissances préalables, ses observations (des accès au SL), et ses objectifs d'inférence (par exemple, ré-identifier ou localiser les utilisateurs). Dans ce cadre, nous considérons l'erreur d'estimation moyenne de l'adversaire comme la bonne métrique pour quantifier le niveau de protection des données de localisation.

Ensuite, nous construisons chaque élément de notre modèle en commençant par la formalisation des profils de mobilité des utilisateurs (comme les connaissances préalables de l'adversaire). Nous faisons usage d'outils statistiques pour générer ces profils, étant donné les traces (passées) des utilisateurs. Les profils des utilisateurs alimentent les attaques d'inférence de la localisation, pour lesquelles nous utilisons principalement des techniques d'inférence bayésienne. Plus particulièrement, nous utilisons des algorithmes d'inférence pour les modèles de Markov cachés afin de dé-anonymiser, de localiser et de reconstruire la trajectoire d'un ou plusieurs utilisateurs. En outre, nous proposons un algorithme générique d'inférence de la localisation basé sur des méthodes de Monte-Carlo (MCMC). Nous fournissons également un outil logiciel : le Mètre de Protection de la Localisation et de la Mobilité (MPL). Sa conception est basée sur notre modèle formel unifié. Il permet d'évaluer l'efficacité de divers mécanismes de protection des données de localisation and de quantifier le niveau de protection des utilisateurs. Par exemple, grâce au MPL, nous validons l'efficacité des méthodes existantes d'embrouillement de la localisation et des mécanismes d'anonymisation sur des trajectoires réelles. Nous démontrons que d'autres métriques (comme k-anonymité et l'entropie) ne sont pas corrélées au succès de l'adversaire (essayant de connaitre les informations privées des utilisateurs), et donc inappropriées. Nos résultats confirment également que l'anonymisation n'est pas une technique adéquate pour protéger les données de localisation. De plus, nos résultats montrent comment la résistance d'un mécanisme de protection varie par rapport aux différents types d'attaques. En conclusion, il est absolument nécessaire d'avoir en tête des objectifs d'attaque concrets lorsque l'on développe des mécanismes de protection des données de localisation.

Finalement, nous développons des techniques optimales d'embrouillement de la localisation contre des attaques de localisation. Un utilisateur a besoin d'un mécanisme de protection de la localisation qui maximise son niveau de protection. L'adversaire, cependant, développe des attaques d'inférence qui minimise son erreur d'estimation. Ces deux objectifs sont en contradiction l'un avec l'autre. Nous proposons donc une méthodologie basée sur la théorie des jeux qui prend en compte à la fois les objectifs de l'utilisateur et de l'adversaire, simultanément. Plus précisément, nous modélisons le problème en utilisant un jeu bayésien de Stackelberg, et le résolvons en utilisant de l'optimisation linéaire. Dans le problème d'optimisation considéré, les utilisateurs imposent au mécanisme de protection de respecter un certain niveau de qualité de service. Nos résultats indiquent que l'anticipation des attaques d'inférence et la prise en considération des connaissances de l'adversaire permettent la réalisation de mécanismes de protection plus efficaces.

Cette thèse est un pas vers une manière plus systématique de modéliser, analyser, et développer des technologies qui améliorent la protection des données (de localisation) privées. Nous proposons différentes méthodes analytiques ainsi qu'un outil logiciel permettant une meilleure comparaison des différents mécanismes de protection. La thèse ouvre également la voie à des investigations futures. Nous sommes convaincus que des approches analytiques similaires peuvent être utilisées pour quantifier et protéger la sphère privée dans des domaines qui n'ont pas été abordés dans cette thèse.

**Mots-clés** : protection des données de localisation, réseaux mobiles, services de localisation, attaques d'inférence, mécanismes de protection des données de localisation, métrique de protection des données privées, inférence bayésienne, modèles de Markov cachés, théorie des jeux, jeu de Stackelberg bayésien.

# Acknowledgements

I would like to thank all who inspired me with their wise words and works, aroused my curiosity with the light of their knowledge, and nourished my creativity with their open minds and fresh souls. I am deeply indebted to many wonderful teachers, colleagues, and friends with whom I traveled the long twenty-year journey, so far, of my education.

I am deeply indebted to my PhD thesis advisor, Prof. Jean-Pierre Hubaux, for giving me the opportunity to work in the stimulating environment of LCA1, for supporting me throughout the PhD process, and for his appreciation of fundamental research and creative thinking. Going back in time, I am also grateful to my master's thesis advisor, Prof. Naser Yazdani, for allowing me to join his wonderful Router Laboratory at the University of Tehran, and for supporting me during my master's studies. Moreover, I owe my gratitude to my undergraduate supervisors, Prof. Naser Movahedinia and Prof. Behrouz Tork Ladani, who walked me into the interesting worlds of computer networks and security.

I would like to thank my committee members for having accepted to assess my thesis, taking the time to read it through, and for all their valuable comments and encouraging words: Dr. George Danezis, Prof. Jean-Yves Le Boudec, and Prof. Vitaly Shmatikov. I would also like to thank Prof. Matthias Grossglauser for being the president of the jury.

I am indebted to all my great collaborators Vincent Bindschaedler, George Danezis, Claudia Diaz, Julien Freudiger, Mathias Humbert, Murtuza Jadliwala, Amir Nayyeri, Panos Papadimitratos, Pedram Pedarsani, Marcin Poturalski, Maxim Raya, Carmela Troncoso, and Maysam Yabandeh for all I learnt through working with them and for all their valuable contributions. Special thanks go to Prof. Jean-Yves Le Boudec, and George Theodorakopoulos for all our fruitful discussions. I am grateful to Mathias for kindly translating the abstract of this thesis into French. I would also like to thank all my students, notably Vasileios Agrafiotis, Vincent Bindschaedler, Quentin Hounkpatin, Ehsan Kazemi, Hai Ly Hoang, Antoine Parisod, Loic Pfister, Gael Ravot, Saeid Sahraei, Francisco Santos, and Ypatia Tsavliri for their great work. I sincerely enjoyed working with all these wonderful people. I am also thankful to all the members of our LCA lab at EPFL and Router Laboratory at the University of Tehran for providing such friendly environments.

I would like to extend my warmest gratitude to my best friends Amir, Maysam, Sajjad, Pedram, Arash, Hossein, Omid, Mahdi, Javad, and Amin for spending so much great time together, for our discussions about many interesting topics on cinema, photography, music, literature, and science, and for your beautiful brilliant minds and elegant insights. I am also thankful to Wojciech, Marcin, and Michal for all our adventurous and enjoyable biking trips in the marvelous Swiss mountains.

Above all, I sincerely thank my loving parents, my gentle brothers, and my beloved wife.

# Contents

# Introduction

*The endless cycle of idea and action,*
*Endless invention, endless experiment,*
*Brings knowledge of motion, but not of stillness;*
*Knowledge of speech, but not of silence;*
*Knowledge of words, and ignorance of the Word.*
*All our knowledge brings us nearer to our ignorance,*
*All our ignorance brings us nearer to death,*
*But nearness to death no nearer to God.*
*Where is the Life we have lost in living?*
*Where is the wisdom we have lost in knowledge?*
*Where is the knowledge we have lost in information?*

T.S. Eliot

Privacy is an absolute essential for individuals' personal development and for the establishment of trust among members of a society. It is valued by individuals, respected in all societies, and recognized internationally as a human right.[1] However, the fast and profound progress in information and communication technologies has increased the vulnerability of individuals' privacy in various domains.

## Privacy Threats

An increasing number of people interact with various computer and communication systems in their everyday lives. Each interaction produces data about, for example, how, when, where, by whom, with whom, and for what purpose these interactions happen. The data that becomes available about individuals is becoming more personal, as most people are now equipped with smart phones with many sophisticated sensors and actuators closely related to their activities. Each of these devices is usually equipped with high-precision localization capabilities based, for example, on a GPS receiver or on triangulation with nearby base stations or access points. In addition, the environment is more and more populated by sensors and smart devices, with which smart phones interact. Operating systems of recent mobile phones also enable multiple personalized applications (from different providers) to constantly run and observe the users' behavior. Many of the services provided by these smart-phone applications need (or request) to have access to users' location data [FDW12]. The use of these personal communication devices, although providing convenience to their owners, exposes an almost indelible digital trace of their whereabouts to the service providers. Note that a location trace is not only a

[1] "No one shall be subjected to arbitrary interference with his privacy, family, home or correspondence, nor to attacks upon his honor and reputation. Everyone has the right to the protection of the law against such interference or attacks." Article 12, Universal declaration of human rights, United Nations, 1984.

set of positions on a map. The contextual information attached to a trace tells much about individuals' habits, interests, activities, and relationships.

Advanced data storage facilities empower governments and corporations to profile an increasing number of users and to keep track of their data traces for long periods of time. This tracking is performed for various financial, strategic, or security reasons, or basically in order to provide services to their users. Moreover, as the cost of data storage tends to zero, and because users' data might have some value at some future time, their data is not deleted. The tools required to analyze (location-tagged) data traces have also made tremendous progress: sophisticated data mining algorithms can leverage on fast growing storage and processing power, thus facilitating the joint analysis of multiple data-sets in parallel.

As the location-tagged data observed from mobile users increase and the cost of storing and processing data decreases, the negative side-effects of insufficient privacy are becoming more and more threatening. Data-holders can learn users' personal or corporate secrets, by using various inference algorithms. Consequently, this can expose people to unwanted advertisements and location-based spams/scams; it can damage their social reputation; it can stimulate service providers to stealthily discriminate among their users (clients); and, it can even make users victims of blackmail or physical violence [Gut02, PC02]. More importantly, information disclosure breaks the balance of *power* between the informed entity (governments and corporations) and the individuals about whom this information is disclosed. What makes the problem of protecting privacy more challenging is that these entities, who gain power and make profit, have little (if any) incentive to change their strategies and limit their user profiling activities. Hence, in order to preserve mobile users' privacy, we need to protect their private data from being unnecessarily disclosed to untrusted entities.

## Protecting Privacy

An individual has *privacy* with respect to an entity if the link between the individual and her personal information is concealed from that entity. The first step towards protecting privacy is to educate users to expose less personal information while interacting with information and communication technologies. Moreover, legal means, as well as privacy enhancing technologies (PETs), need to be designed and used to preserve the privacy of users in their use of computer and communication systems. PETs limit the amount of information that can be leaked about users in these systems. To further enforce privacy protection, legal means can prevent the organizations from misusing the information that users share with them, and can encourage them to protect their users' privacy. In this thesis, we address the approach of using privacy enhancing technologies.

In order to better explain privacy enhancing technologies, let us assume some users communicating with each other and with some servers over a network. In this setting, users' personal data is placed in different places: in the computer of the users themselves, in the communication network (e.g., in routers), and in the servers (e.g., in the database of a recommender system or a social networking service provider). Privacy enhancing technologies have been proposed to defend users' privacy in these three (user, network, server) fronts.

The data located on the users' side can be obfuscated before they leave the users' computers (e.g., using data perturbation techniques [AA01, AS00], network perturbation [MPS13, SPTH09], or traffic obfuscation [WCM09]). This prevents an observer to have a correct view on users' data. Furthermore, the users' sensitive information, such as their IP addresses

or address of their destinations, can be concealed in the communication network (e.g., using anonymity[2] networks [Cha81, DDM03, DMS04, FM02, GRS99, RR98]). In online interactions, the identity of a user can be concealed using anonymous credentials [Bra00, CVH02]. Users' privacy can be protected by distributing their data over multiple entities that do not collude with each other (e.g., P2P social networks [BSVD09, CMS09], and distributed search engines [Yac]). Some trusted entities can be introduced, or communities can be formed, to process users' data and provide only the aggregated information to the servers (e.g., community-based recommender systems [Can02], and anonymity proxies). These entities can even distort the aggregated results to further protect users' privacy, for example using mechanisms that provide differential privacy [Dwo06, Dwo08, DMNS06, MT07]. Last but not least, users and servers can agree to run a cryptographic protocol that limits what any observer (including the servers) can learn about the users. The server gains only the information that it needs in order to provide the service (e.g., private information retrieval [CGKS95, CKGS98], private data mining [DZ03, LP00]). A combination of these defense mechanisms can be used to further enforce users' privacy.

Effectiveness of these PETs are evaluated in different ways. For example, information theoretic measures (e.g., entropy and channel capacity [CT06, Sha48]) and statistical methods (e.g., Metropolis-Hastings algorithm) are used to quantify how much anonymity a privacy preserving mechanism provides [CPP08, Dan04, DSCP02, SD02, Tro11]. By analyzing traffic and topology of networks (of computers or users), it is shown that anonymization alone does not necessarily protect users' privacy. In fact, given some auxiliary (background / side-channel) information about the users and the properties of their protection mechanisms, an adversary can break/reduce their anonymity [HVCT10, MD05, NS08, NS09, NS10]. These findings quest for PETs that anticipate the adversary's knowledge and attacks.

PETs need also to take the system specifics and users' requirements into account. Despite the strength of various PETs, not all of them can be applied in all scenarios due to their specific requirements or restrictions. Data perturbation and traffic obfuscation can reduce users' service quality. So, they cannot be used when the service quality cannot be sacrificed (e.g., checking if someone is susceptible to cancer from her genomic data). Anonymity networks require volunteers to run the protection mechanism and many participants to use it in order to provide an acceptable level of privacy. So, it is less effective when used by a few users in a small network. Output distortion techniques (used in providing differential privacy), must be applied on the aggregated data of multiple users. Therefore, it is not suitable for the case where each single user accesses the server directly. Cryptographic approaches require changes on the side of users and the servers. Consequently, they cannot be used against untrusted service providers who do not have (economic) incentives to change their system models.

## Protecting Location Privacy

Smart phone users in mobile networks connect to the Internet and, similarly to the desktop users, participate in activities such as web surfing and online social networking. Thus, similar privacy concerns and protection mechanisms exist in mobile networks as well. However, what makes mobile users different from desktop users is the heavy usage of location-based services (LBSs). In this thesis, we focus on analyzing and protecting users' *location privacy*, which is more specific to mobile networks.

[2] Anonymity is the state of being indistinguishable among a set of users (anonymity set) [PK08].

The users' location information that is shared with various location-based services enables the service providers (or any untrusted observer) to identify the users' personal (or regularly visited) locations, such as their homes and workplaces. This information can be used to identify the users, even if they are anonymous/pseudonymous in the network layer [FSH11, GP09, HGXA06, Kru07, MYYR10, DMDBP08, ZB11]. To overcome this problem, users can use temporal pseudonyms [BS04, BHV07, CPHL07, FMHP09, JWH07, LSHP06], hide their location occasionally [BS04, HGH$^+$08, HGXA07, HYMS05, HYMS06, JWH07, LSHP06], report a noisy version of their location [GG03, MRC09], hide their reported location in a large area [BLPW08, CML06, GL08, GG03], or report fake locations [CG09, KYS05, Kru09a, LJY08, YPL07] when connecting to an LBS. Cryptographic tools can be used, in addition to obfuscation mechanisms, in order to allow users to obtain desired location-based services while hiding their true locations from the service providers [NTL$^+$11, FPIU10, PBP10, ZGH07].

Different protection mechanisms have been evaluated with different location privacy metrics. Two popular examples are k-anonymity [Swe02] and entropy [DSCP02, SD02]; they have been adapted to measure location privacy in the case of sporadic access to the LBS [BWJ05, GL05, GL08, GG03, KGMP07, MCA06, SHL$^+$05]. In the case of continuous location updates, the uncertainty of an adversary in linking subsequent location updates of a user [BS04, FMHP09, HGH$^+$08, HGXA07, HYMS05, HYMS06, JWH07, LSHP06], or the k-anonymity of the user at the trace level [BWJ05, GDV08, NAS08, XC08] have been used to quantify users' location privacy.

Notwithstanding the many contributions from different disciplines (such as databases, mobile networks, and ubiquitous computing) for protecting location privacy, the lack of a unified and generic formal framework for specifying protection mechanisms and for evaluating location privacy is evident. This has led to the divergence of contributions, hence, it has caused confusion about which mechanisms are more effective. The adversary model is often not appropriately addressed and formalized, and a good model for the knowledge of the adversary and his possible inference attacks is missing. Additionally, the privacy metrics are not properly correlated with the information that the adversary gains in an inference attack. These can unfortunately lead to a wrong estimation of the location privacy of mobile users, both for evaluation and protection purposes.

## Contributions and Outline of the Thesis

In this thesis, we propose an analytical framework for the location privacy of LBS users. In such a setting, users share their location plus complementary application-dependent information (e.g., their opinion about the location) with the LBS. However, users are concerned about any third-party observer who tracks their shared locations and therefore violates their privacy. We assume users use privacy enhancing technologies that do not require them to change the architecture of LBS (who has little incentive to cooperate), or trust on third-party entities such as anonymity proxies (which does not solve the problem, but shifts it to an entity other than LBS). Furthermore, as users independently connect to LBS and expect immediate response, data aggregation and output perturbation mechanisms are not very applicable neither. So, the location-privacy protection mechanisms that we consider in this thesis are in the category of data perturbation mechanisms. In other words, instead of reporting their true and precise location, users report a pseudo-location (an obfuscated location) to the LBS at each (pseudonymous) access. In our framework, we make explicit assumptions about the ad-

versary's knowledge, observation, and inference objectives. We define users' mobility model, their access pattern to location-based services, and their privacy and service-quality requirements. Relying on this framework, we provide methodologies to quantify and protect location privacy of mobile users who make use of different types of location-based services.

More precisely, the contributions of this thesis are as follows:

**Chapter 1** We propose a *framework* that encompasses the main elements and entities that affect the location privacy of mobile users. We provide probabilistic models for the mobility of users, their access patterns to LBSs, the location-privacy preserving mechanisms (LPPMs), and location-inference attacks. LPPMs anonymize and obfuscate users' locations before being observed by an untrusted entity (adversary). We show how different types of location obfuscation methods, such as location perturbation, location hiding, and adding dummy locations, can be specified in our formalism. In our framework, we also define the adversary's knowledge on users' mobility patterns. Furthermore, we define the adversary's inference objectives. For example, he might be interested in re-identifying the users from whom anonymous traces are observed, finding the location of a specific user at a given time, or estimating the number of times that two users have been in each other's proximity. We suggest metrics for quantifying users' location privacy gain and service quality. By formalizing the adversary's inference attack as an estimation problem, we distinguish between accuracy, certainty, and correctness of his estimation. We then argue that the estimation error of the adversary (i.e., his incorrectness) is the appropriate metric to quantify (location) privacy. Our related publications, which contribute to this framework, are [SFH10, SFJH09, STD+11, STT+12, STLBH11].

**Chapter 2** We formalize the adversary's background *knowledge* on users' mobility patterns. In this thesis, we formalize users' mobility, i.e., the relation between each user and locations over time, as a (time-period dependent) first order Markov chain. We assume that the adversary has some prior information about the users' locations. For example, he might know where they live and work, or where their locations of interests are. He might also have access to some location traces of the users. We propose a Bayesian inference *user profiling* algorithm for the adversary. The profiling algorithm turns the adversary's prior information on users' traces into his background knowledge on their mobility. To compute the users' profiles through the Bayesian inference process, we make use of a Monte-Carlo method called Gibbs sampling. The outcome of this iterative method converges to users' mobility patterns given the prior information. Our related publication is [STLBH11].

**Chapter 3** We provide a generic model that formalizes the adversary's *attacks* against the private location-information of mobile users. In particular, we rigorously define and provide inference methods for de-anonymization, tracking, and localization attacks on anonymous traces. An adversary who knows users' mobility profiles and observes their distorted and anonymous location traces, tries to infer what has been hidden from him. We rely on well-established statistical methods to evaluate the performance of such inference attacks. More precisely, as we assume a Markovian model for the users' mobility, we use hidden Markov models to invert location-privacy preserving mechanisms. We formalize the adversary's success and we quantify location privacy as the adversary's expected estimation error. We also (quantitatively) show the inappropriateness of some

existing metrics, notably entropy and k-anonymity, for quantifying location privacy. Our related publications are [SFH10, SFJH09, STD+11, STLBH11, STD+10].[3]

**Chapter 4** We provide a methodology for designing optimal *location-privacy preserving mechanisms* against inference attacks. On the one hand, privacy-concerned mobile users make use of LPPMs to increase their location privacy with respect to untrusted observers (including the service provider). On the other hand, the adversary (the untrusted observer) tries to infer more information about users' locations. These two objectives are obviously in conflict, and optimizing one damages the other. Hence, a game naturally emerges between users and the adversary. We formalize the problem of designing optimal LPPMs for users as a Bayesian Stackelberg game, where users anticipate the best inference attacks on their defense mechanism. We solve the game for each user against an adversary who tries to localize users by observing their sporadic accesses to an LBS. We design linear programs to find an optimal strategy for each user, respecting her service quality constraints and location privacy sensitivities. Subsequently, we find the optimal inference attack for the adversary. Our related publication is [STT+12].[4]

**Software** We devoted considerable time and effort to designing and developing a software tool named *Location-Privacy and Mobility Meter* (LPM). We implemented all the elements of our framework in this tool. Our user profiling algorithm, and location inference attacks are also included. LPM is a modular software that enables the comparison of various LPPMs in terms of their effectiveness in protecting users' location privacy. It also enables us to analyze users' mobility models. For example, we can compute the randomness of a user's mobility, or the similarity between two users' mobility models. LPM allows us to define time periods, location regions, and the user-specific privacy and service quality distance metrics. Given this configuration, we can run multiple attacks on multiple LPPMs and compare the users' location privacy for each attack-defense pair. LPM is an object-oriented tool and is implemented in C++. Its fully documented code, plus examples and a quick start guide, are available online [LPM12].

In summary, the main contribution of this thesis to the field of location privacy is twofold: (i) we consider the adversary model as the inseparable element in analysis and design of privacy preserving mechanisms, and (ii) we provide theoretical methods plus software tools to analyze and design privacy preserving mechanisms.

[3] Our related publication, which is not covered in this thesis, is [FSH11]. In this paper, we propose methods to identify personal locations (e.g., home and workplace) of mobile users. We show that this location identification can lead to the identification of users even if they contact the LBS anonymously and sporadically (but, over an long enough time period). Our results emphasize the need to protect users' location privacy.

[4] Our related publications, which are not covered in this thesis, are [SHSH11, FSH09, RSH10, SPTH11, SPTH09]. In [SHSH11, SPTH11], we propose a collaborative approach to protect location privacy of mobile users relaying on wireless peer-to-peer communication. The idea is that a user can share the location-based content that she obtains from the LBS with other users. So, users can hide from the LBS while receiving the contextual information they seek. In [SPTH09], we propose another collaborative approach that enables users of recommender systems to preserve their privacy with respect to the service provider, and still obtain accurate recommendations. The idea is that users can inject fake ratings, obtained from other similar users, into their profiles. In [FSH09], we explain how mix zones (where mobile users remain silent and change their pseudonym) can be placed in an area to optimize users' location privacy. In [RSH10], we discuss the conflict between preserving privacy and establishing trust among mobile users.

# 1 A Unified Framework

There are various elements that influence the location privacy of smart phone users, especially while various location-based services are being used. As these components are highly interconnected, they should be studied together in a consistent framework. Such a framework should consist of the following: (1) the structure and the semantic of the underlying location map within which considered users move, (2) a model representing users' mobility and the regularity of their activities, (3) a model of the location-based services and of the users' private information leakage pattern through such services, (4) an adversary model representing the threats against users' location privacy, (5) a model of location-privacy preserving mechanisms, (6) a metric for measuring the service quality loss that users incur by using a protection mechanism, and (7) an evaluation metric for quantifying the location privacy of users, given a protection mechanism against a given inference attack. In this chapter, we build a unified framework and we briefly describe these components and their interrelation. This framework allows us to precisely define location privacy and to evaluate and compare the cost/benefit of various location-privacy preserving mechanisms with respect to different attacks. In this chapter, we state our basic assumptions and define the notations and terminology that we use throughout the thesis.

We define a location-privacy framework (system) as a tuple of the following inseparable elements: $\langle \mathcal{U}, \mathsf{LOC}, \mathcal{A}, \mathsf{LBS}, \mathsf{LPPM}, \mathsf{SQ}, \mathcal{O}, \mathsf{ADV}, \mathsf{PRV} \rangle$, where $\mathcal{U}$ is the set of mobile users who move within an area whose geographical and contextual information is embodied in $\mathsf{LOC}$, and $\mathcal{A}$ represents the set of possible location traces for the users, in a given time window. $\mathsf{LBS}$ is the location-based service[1] through which users share (expose) their location to the service provider (and potentially other untrusted entities) and consequently open the door to various privacy threats. Location-privacy preserving mechanism, $\mathsf{LPPM}$, acts on the actual traces $\mathbf{a}$ (which is a member of $\mathcal{A}$) and produces the observed traces $\mathbf{o}$ (a member of $\mathcal{O}$, which is the set of *observable* traces to an adversary $\mathsf{ADV}$). Hence, when accessing the $\mathsf{LBS}$, users only expose the output of $\mathsf{LPPM}$, instead of sharing their actual locations. The users' obtained service quality is measured by an evaluation metric $\mathsf{SQ}$. The adversary $\mathsf{ADV}$ is an entity who implements inference (reconstruction) attacks to infer some information about $\mathbf{a}$ (e.g., location of a given user at a given time instant), having observed $\mathbf{o}$ and by relying on his knowledge of the $\mathsf{LBS}$ and $\mathsf{LPPM}$ and of the users' mobility model (profile). The performance of the adversary and his success in recovering the desired information about $\mathbf{a}$ is captured by an evaluation metric $\mathsf{PRV}$. The success of the adversary and the location-privacy of users are two sides of the same coin, which are coupled together using $\mathsf{PRV}$. In the following sections, we present and specify all the entities and components of our framework and illustrate their inter-relationships.

Figure 1.1 shows a sketch of our framework. Table 1.1 presents the summary of notations that we introduce here and use in the next chapters. Throughout the thesis, we use bold italic

[1] We use this term for referring to location-sharing services too.

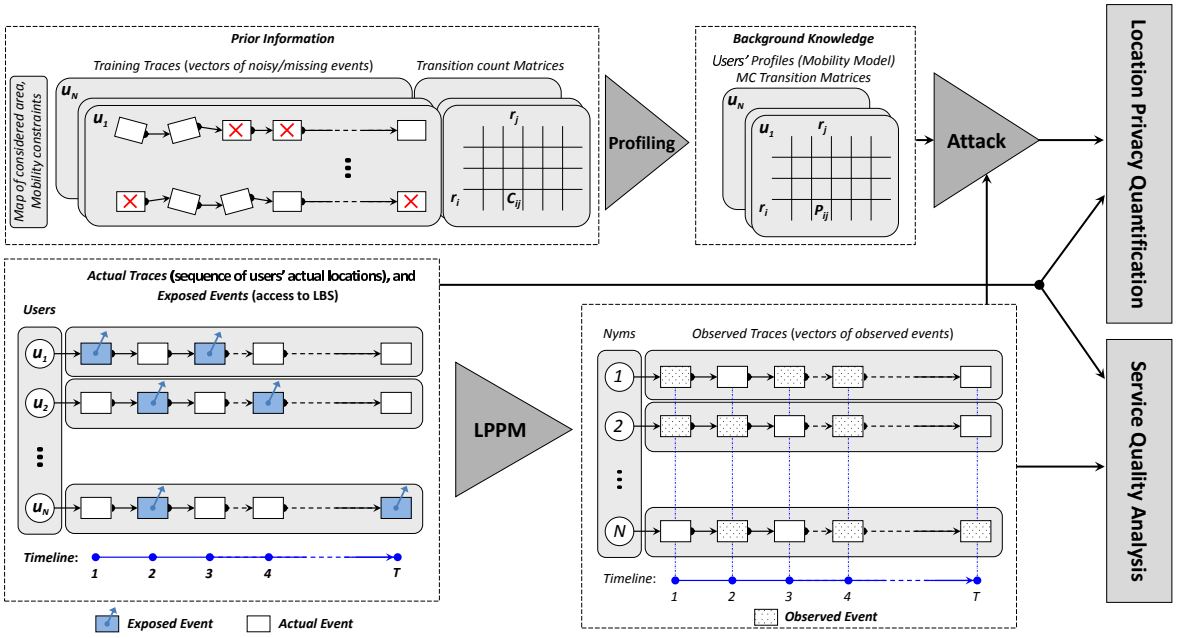



Figure 1.1: A unified framework for location privacy. Time and location are discrete. Users access (expose themselves to) the LBS at different times. Location-privacy preserving mechanisms (LPPMs) perform access anonymization and location obfuscation for users. So, users share their anonymous and obfuscated locations with LBS. The attacker processes a set of available location traces associated with users in order to create a mobility profile for each one. A profile is in the form of transition probability matrix of a Markov chain. Given the users' mobility profiles and the observed traces, the adversary tries to reconstruct (infer) the actual traces. Location privacy of users and the service quality that they experience is computed given the users' actual traces, LPPM outcome, and the result of the attack.

capital letters ($\boldsymbol{X}$) to denote random variables, bold lower case letters to denote vectors and matrices ($\mathbf{x}$), the script letters ($\mathcal{X}$) to denote sets, and lower case letters ($x$) to denote the elements of vectors or members of sets. For simplicity, we sometimes use briefer notation if it is clear from the context; for example, we might use $\Pr\{\mathbf{a}\}$ to mean $\Pr\{\boldsymbol{A} = \mathbf{a}\}$.

## 1.1 Users, Time, and Space

We consider $\mathcal{U} = \{u_1, u_2, \ldots, u_N\}$ a set of $N$ mobile users who move within an area that is partitioned into $M$ distinct regions (locations) $\mathcal{R} = \{r_1, r_2, \ldots, r_M\}$. Time is also considered to be discrete, and the set of time instants when the location of users might be observed is $\mathcal{T} = \{1, \ldots, T\}$.[2]

We can represent the information associated to each location in a three-layer model. The first layer, to which we refer the most, models the *geographical coordinate space* of the region. The second layer models the distinguishable *places* (e.g., hospital, bus station) in the region, and the third layer models the *semantic* associated to the places, which determines the type of activities that happen in the region.

[2] The set of time instants in $\mathcal{T}$ can be partitioned by some time periods (e.g., morning, afternoon, evening, night).

The precision at which we want to represent the user mobility determines the granularity of the space and time. For example, if the mobility is supposed to have a low/high precision, regions can be of a city/block size, and two successive time instants can be a day/hour apart.

### 1.1.1 Events and Traces

The spatio-temporal position of users is modeled through events and traces. An *event* is defined as a triplet $\langle u, r, t\rangle$, where $u \in \mathcal{U}$, $r \in \mathcal{R}$, $t \in \mathcal{T}$. A *trace* of user $u$ is a $T$-size vector of events $\mathbf{a}_u = \begin{bmatrix} a_u^1 & a_u^2 & \cdots & a_u^T \end{bmatrix}$. The set of all traces that belong to user $u$ is denoted by $\mathcal{A}_u$. Notice that, of all the traces in $\mathcal{A}_u$, exactly one is the true trace that user $u$ created in the time window of interest $\mathcal{T}$; the true trace, denoted by $\mathbf{a}_u$, is called the *actual trace* of user $u$, and its events are called the *actual events* of user $u$. The set of all possible traces of all users is denoted by $\mathcal{A} = \mathcal{A}_{u_1} \times \mathcal{A}_{u_2} \times \ldots \times \mathcal{A}_{u_N}$; the member of $\mathcal{A}$ that is actually created by the $N$ users is denoted by $\mathbf{a}$ and it is equal to

$$\mathbf{a} = \begin{bmatrix} a_{u_1}^1 & a_{u_1}^2 & \cdots & a_{u_1}^T \\ a_{u_2}^1 & a_{u_2}^2 & \cdots & a_{u_2}^T \\ \vdots & \vdots & \ddots & \vdots \\ a_{u_N}^1 & a_{u_N}^2 & \cdots & a_{u_N}^T \end{bmatrix}. \tag{1.1}$$

### 1.1.2 User Mobility Model

The matrix of all actual events $\mathbf{a}$ is in fact a sample from the random variable $\boldsymbol{A}$ that is distributed according to $\Pr\{\boldsymbol{A} = \cdot\}$. This distribution reflects the joint mobility pattern of the users. We refer to each marginal distribution $\Pr\{\boldsymbol{A}_u = \cdot\}$ as the *mobility profile* of user $u$, that is $\mathbf{a}_u \sim \Pr\{\boldsymbol{A}_u = \cdot\}$. In this work, we assume that the users' mobility profiles are independent of each other, i.e., $\Pr\{\boldsymbol{A} = \mathbf{a}\} = \prod_u \Pr\{\boldsymbol{A}_u = \mathbf{a}_u\}$. In other words, the location of a user is independent of others, *given* the user's mobility profile (i.e., there is a *conditional independence* between the users' locations).

As users tend to have different mobility patterns at different *time periods* (e.g., morning vs. afternoon, or weekday vs. weekend), we assume the users' mobility profiles to be time-period dependent. Because of the time dependence, we take into account indirect correlation among the users' locations, for instance traffic jams in the morning and in the evening.

We assume that the mobility of a user, in each time period, is modeled as a Markov chain on the set of regions. So, for user $u$, the probability distribution of actual traces can be expressed using the transition matrix of its Markov chain. Each state of the Markov chain represents a region. We use $p_{r,r'}(u)$ to indicate the probability of a transition from region $r$ to $r'$ by user $u$. We also use $\pi_r(u)$ to indicate the probability that user $u$ is in region $r$. In Chapter 2, we propose a method to construct the users' mobility profiles.

The Markov chain model can be turned into a more powerful (yet more complex) model, depending on how the states of the chain are defined. If states, or the transition between states, represent complex previous location visits (e.g., past $k > 1$ locations in the past), or the user's current/previous activity, then the model can become arbitrarily accurate.

| | |
|---|---|
| $\mathcal{U}$ | Set of mobile users $u_1, u_2, ..., u_N$. |
| $\mathcal{R}$ | Set of regions (locations) $r_1, r_2, ..., r_M$ that partition the whole location area. |
| $\mathcal{T}$ | Time window under consideration (when users make use of a LBS and their locations can be observed). Without loss of generality, we assume $\mathcal{T} = \{1, 2, ..., T\}$. |
| $a = \langle u, r, t \rangle$ | An actual location event: User $u \in \mathcal{U}$ at time $t \in \mathcal{T}$ is in location $r \in \mathcal{R}$. |
| $a_u^t$ | Location of user $u$ at time $t$. |
| $\mathbf{a}_u$ | Actual trace of user $u$. |
| $\mathbf{a}$ | Actual traces of all users. |
| $\mathcal{A}$ | Set of all possible actual traces (of all users). Accordingly, $\mathcal{A}_u$ is the set of possible actual traces of user $u$. An actual trace $a$ is a member of $\mathcal{A}$ that is generated probabilistically according to the random variable $\boldsymbol{A}$. |
| $\mathbf{x}$ | LBS access pattern of users. The vector $\mathbf{x}_u \in \{0,1\}^T$ reflects the time instants when user $u$ accesses the LBS. Set $\mathcal{X}$ is the set of possible LBS access patterns, where $\mathbf{x}$ is its member generated probabilistically according to random variable $\boldsymbol{X}$. |
| $\widetilde{\mathcal{U}}$ | Set of user pseudonyms $\tilde{u}_1, \tilde{u}_2, ..., \tilde{u}_{N'}$. |
| $\widetilde{\mathcal{R}}$ | Set of pseudolocations $\tilde{r}_1, \tilde{r}_2, ..., \tilde{r}_{M'}$. |
| $\sigma(u)$ | Persistent pseudonym of user $u$. |
| $\boldsymbol{\sigma}(u)$ | Pseudonyms of user $u$ over time. Element $\sigma^t(u)$ is the user's pseudonym at $t$. |
| $\boldsymbol{\sigma}$ | Pseudonyms of all users. It is generated according to random variable $\boldsymbol{\Sigma}$. |
| $o = \langle \tilde{u}, \tilde{r}, t \rangle$ | An observed location event, representing that some user with pseudonym $\tilde{u} \in \widetilde{\mathcal{U}}$ at time instant $t \in \mathcal{T}$ is associated with pseudolocation $\tilde{r} \in \widetilde{\mathcal{R}}$. We use $o_{\tilde{u}}^t$ to refer to the pseudolocation $\tilde{r}$ in this observed event. Vector $\mathbf{o}_{\tilde{u}}$ is an observed trace with pseudonym $\tilde{u}$. Matrix $\mathbf{o}$ represents observed traces of all users, that is generated probabilistically according to the random variable $\boldsymbol{O}$. |
| $\mathcal{O}$ | Set of possible observable traces from all users. |

Table 1.1: Table of notations. We use bold italic capital letters ($\boldsymbol{X}$) to denote random variables, bold lower case letters to denote vectors and matrices ($\mathbf{x}$), the script letters ($\mathcal{X}$) to denote sets, and lower case letters ($x$) to denote the elements of vectors or members of sets.

## 1.2 Location-based Services

Smart phones, among other increasingly powerful mobile computing devices, offer various methods of localization. Integrated GPS receivers, or positioning services based on nearby communication infrastructure, enable users to position themselves fairly accurately. This gives rise to a range of *Location-based Services* (LBSs) that rely on the users' current location: users can connect to a LBS server and share/obtain contextual information relevant to their current location and surroundings.[3]

Let $\mathbf{x}_u \in \{0,1\}^T$ be a vector that shows the LBS access times of user $u$ over the time $\mathcal{T}$. In effect, $\mathbf{x}_u$ acts as a bit-mask, for example, if $x_u^t = 1$, then user $u$ sends a query to the LBS that contains her actual location at time $t$, i.e., $a_u^t$ is exposed to the LBS. Thus, $\mathbf{x}_u$ reflects the access pattern of user $u$ to the LBS.

We model an LBS application (that runs on the user's smart phone) as a function that

[3] This includes location-sharing services as well. However, for the sake of simplicity, throughout this thesis, we just refer to them as LBSs.

maps actual traces $\mathbf{a} \in \mathcal{A}$ to a random variable $\boldsymbol{X}$ that takes values in the set $\mathcal{X} = \{0,1\}^{N \times T}$. The corresponding probability distribution function $\Pr\{\boldsymbol{X} = \mathbf{x} \,|\, \boldsymbol{A} = \mathbf{a}\}$ can be computed as follows, considering that mobile users usually make use of the location-based services at each time instant, independently from each other:

$$\Pr\{\boldsymbol{X} = \mathbf{x} \,|\, \boldsymbol{A} = \mathbf{a}\} = \prod_u \prod_t \Pr\{\boldsymbol{X}_u^t = x_u^t \,|\, \boldsymbol{A}_u^t = a_u^t\}. \tag{1.2}$$

### 1.2.1 Sporadic vs. Continuous

We differentiate among location-based services according to the frequency at which users access them. On one end of the spectrum, there are LBSs that require their users to continuously access them, whereas on the other end, there are the majority of LBSs whose users access them in a rather sporadic manner, i.e., there is a non-negligible time gap between two successive accesses of a user to the LBS. In other words, an application is considered to be *sporadic* if the exposed locations from the users are sparsely distributed over time, and it is considered *continuous* if they are very close in time. Most of the existing location-based services, such as Foursquare[4] and various nearby points-of-interest search services, are sporadic as they do not need their users to share their location all the time.

## 1.3 Location-Privacy Preserving Mechanisms

Mobile users share their location with possibly untrusted entities in various location-based services, or unwillingly expose their location to curious eavesdropping entities through the wireless channel. In all these scenarios, an adversarial entity can track users over an observation period, unless their actual traces are properly modified and distorted before being exposed to others, i.e., before becoming observable. The mechanism that performs this modification in order to protect the users' location-privacy is called a Location-Privacy Preserving Mechanism (LPPM).

LPPMs can be implemented in different architectures: *centralized* vs. *distributed*. The protection can be performed in the centralized architecture by a trusted third party (mostly known as the central anonymity server or privacy proxy) as opposed to being done by the users or on their mobile devices in a distributed architecture, where modifications are (mostly) done independently from each other, i.e., in a *user-centric* manner. Next, we abstract away these details and provide a generic model for LPPMs.

A location-privacy preserving mechanism receives a set of actual traces $\mathbf{a}$ and their corresponding LBS access traces $\mathbf{x}$, one for each user, and modifies them in two steps: *obfuscation* and *anonymization*.

### 1.3.1 Obfuscation

In the obfuscation process, the location-stamp of each exposed event can be obfuscated, i.e., replaced by a *pseudolocation* in the set $\widetilde{\mathcal{R}} = \{\tilde{r}_1, ..., \tilde{r}_{M'}\}$.[5] Each pseudolocation corresponds to a subset of regions in $\mathcal{R}$. Hence, $\widetilde{\mathcal{R}} = \mathcal{P}(R)$, and $M' = 2^M$. In the case when the user does

[4] https://foursquare.com/

[5] Note that, in this work, we do not consider time obfuscation.

not access the LBS (hence her actual location is not exposed), a fake pseudolocation can be produced by the LPPM. This is equivalent to saying that the LPPM selects a fake location for the user and then obfuscates it. According to our definition of location obfuscation, hiding a location can be done by replacing it with the pseudolocation {}, perturbing a location can be implemented by replacing it with another single location (e.g., by adding a noise to the actual location), and generalizing a location can happen by replacing it with a pseudolocation that refers to a set of locations (e.g., by reducing the granularity of the location). Notice that each location can be obfuscated to a different pseudolocation each time it is encountered.

### 1.3.2 Anonymization

In the anonymization/pseudonymization process, the user identity part of each event is replaced by a *user pseudonym* in the set $\widetilde{\mathcal{U}} = \{\tilde{u}_1, ..., \tilde{u}_{N'}\}$. Users might change their pseudonym any time they access the LBS, or keep it for a longer time and have a persistent pseudonym. The possibility of changing pseudonym depends on the LBS, whether it allows implementing a multiple pseudonym scheme. Let $\sigma^t(u)$ be the pseudonym of user $u$ at time $t$, which will be equal to $\sigma(u)$ in case of persistent pseudonyms. We denote by $\sigma^t(u) = \tilde{u}$ the assignment of pseudonym $\tilde{u}$ to user $u$ at time $t$. Let $\boldsymbol{\sigma}_u$ be the pseudonym vector of user $u$ over time, and $\boldsymbol{\sigma}$ be the matrix of pseudonyms for all users during the observation time. An assignment of pseudonyms to users, $\boldsymbol{\sigma}$ is selected according to the random variable $\boldsymbol{\Sigma}$.

For the case of persistent pseudonyms, each user's pseudonym remains the same for the whole observation time $\mathcal{T}$. In this thesis, the *anonymization* mechanism that we consider, is the random permutation. That is, a permutation of the users is chosen uniformly at random among all $N!$ permutations and each user's pseudonym is her position in the permutation. More formally, the anonymization mechanism selects, independently of everything else, a permutation $\boldsymbol{\sigma}$ according to the probability distribution function $\Pr\{\boldsymbol{\Sigma} = \boldsymbol{\sigma}\} = \frac{1}{N!}$.

### 1.3.3 Observed Events and Traces

The LPPM, as the combination of the two (obfuscation and anonymization) processes, probabilistically maps exposed traces $(\mathbf{a}, \mathbf{x}) \in \mathcal{A} \times \mathcal{X}$ to obfuscated and anonymized traces. The output is distributed according to a random variable $\boldsymbol{O}$ that takes values in the set $\mathcal{O}$, which is the set of all possible obfuscated and anonymized traces of all users. Such a trace is composed of $T$ events of the form $o = \langle \tilde{u}, \tilde{r}, t \rangle$, where $\tilde{u} \in \widetilde{\mathcal{U}}$, $\tilde{r} \in \widetilde{\mathcal{R}}$, for $t = \{1, 2, \cdots, T\}$. A complete trace, observed from a single user, is denoted by $\mathbf{o}_{\tilde{u}}$. We denote by $\mathbf{o}$ the spatiotemporal position of users over time as perceived by the observer (e.g., LBS), and we call it the *observed* traces of users. In the case of persistent pseudonyms:

$$\mathbf{o} = \begin{bmatrix} o^1_{\tilde{u}_{\sigma(u_1)}} & o^2_{\tilde{u}_{\sigma(u_1)}} & \cdots & o^T_{\tilde{u}_{\sigma(u_1)}} \\ o^1_{\tilde{u}_{\sigma(u_2)}} & o^2_{\tilde{u}_{\sigma(u_2)}} & \cdots & o^T_{\tilde{u}_{\sigma(u_2)}} \\ \vdots & \vdots & \ddots & \vdots \\ o^1_{\tilde{u}_{\sigma(u_N)}} & o^2_{\tilde{u}_{\sigma(u_N)}} & \cdots & o^T_{\tilde{u}_{\sigma(u_N)}} \end{bmatrix}, \tag{1.3}$$

In this work, we mainly study the case where each exposed event of a user is obfuscated independently of other events that belong to that user or other users. The mobility profiles of all users can be used by the LPPM in the process of obfuscating users' locations. This

knowledge of the users' mobility profiles enables us to design strong LPPMs against the adversary who also relies on this type of information (see Chapter 4).

The probability of a given LPPM output $\mathbf{o}$ is then computed as follows, for the case of using persistent pseudonyms:

$$\begin{aligned}
&\Pr\{\boldsymbol{O} = \mathbf{o} \mid \boldsymbol{X} = \mathbf{x}, \boldsymbol{A} = \mathbf{a}\} \\
&= \sum_{\boldsymbol{\sigma}} \Pr\{\boldsymbol{O} = \mathbf{o}, \boldsymbol{\Sigma} = \boldsymbol{\sigma} \mid \boldsymbol{X} = \mathbf{x}, \boldsymbol{A} = \mathbf{a}\} \\
&= \sum_{\boldsymbol{\sigma}} \underbrace{\Pr\{\boldsymbol{O}_{\tilde{u}} = \mathbf{o}_{\tilde{u}} \mid \sigma(u) = \tilde{u}, \boldsymbol{X}_u = \mathbf{x}_u, \boldsymbol{A}_u = \mathbf{a}_u\}}_{\text{Obfuscation Mechanism}} \cdot \underbrace{\Pr\{\boldsymbol{\Sigma} = \boldsymbol{\sigma} \mid \boldsymbol{X} = \mathbf{x}, \boldsymbol{A} = \mathbf{a}\}}_{\text{Anonymization Mechanism}}
\end{aligned} \tag{1.4}$$

For each user $u$ the LPPM chooses a pseudonym $\tilde{u}$ according to the anonymization mechanism, and for each actual event $a = \langle u, r, t \rangle$ the LPPM chooses a pseudolocation $\tilde{r}$ by sampling from the obfuscation probability distribution, hence outputting $o = \langle \tilde{u}, \tilde{r}, t \rangle$.

Notice that, in general, using an LPPM reduces the quality of the information provided to the location-based service. Consequently, the service quality that the user receives is also reduced. Therefore, there exists a tradeoff between the effectiveness of the LPPM and the experienced service quality for the user. We mainly address this tradeoff when designing an optimal LPPM for the users, in Chapter 4.

## 1.4 Adversary Model

The privacy of users in a mobile network can be threatened in different ways. As stated before, an *adversary* can be present in the physical vicinity of the users and eavesdrop their wireless communication with the LBS. He can also be the LBS operator itself, or an entity sitting in the LBS network sniffing the users' location-tagged information. In this thesis, we abstract away how the adversary technically observes users' traces.

In order to accurately evaluate and effectively design an LPPM, we must formalize the adversary against whom the protection is placed. Hence, the adversary model is certainly an important, if not the most important, element of a location-privacy framework. An adversary is characterized by his *knowledge* and *attack(s)*. A framework should provide a model for the adversary knowledge and specify how the adversary obtains and constructs his knowledge, and which attacks he performs in order to reconstruct location information of the users.

### 1.4.1 Background Knowledge

The adversary is assumed to be aware of the type and the characteristics of the location-based service, and also of the location-privacy preserving mechanism. More precisely, he knows the anonymization and obfuscation probability distribution functions of the LPPM and the exposure probability distribution function of the LBS.

The adversary might have access to some (possibly noisy or incomplete) traces of users, and other public information about locations visited by each user, such as their homes and workplaces. We refer to this as his *prior information*. From this information, the adversary constructs a *mobility profile* $(\hat{\mathbf{p}}_u, \hat{\boldsymbol{\pi}}_u)$ for each user $u \in \mathcal{U}$. In Chapter 2, we explain in detail one way of profiling users and constructing the adversary's *background knowledge*.

### 1.4.2 Inference Attacks

Given the LBS and LPPM, the users' mobility profiles $\{(\hat{\mathbf{p}}_u, \hat{\boldsymbol{\pi}}_u)\}_{u\in\mathcal{U}}$ estimated by the adversary, and the observed traces $\mathbf{o}$ that are produced by the LPPM, the attacker runs an inference (reconstruction) attack and formulates his objective as a question of the $\mathcal{U}-\mathcal{R}-\mathcal{T}$ type. Schematically, in such a question, the adversary specifies a subset of users, a subset of regions and a subset of time instants, and he asks for information related to these subsets. If the adversary's objective is to find out the whole sequence (or a partial subsequence) of the events in a user's trace, the attack is called a *tracking* attack. The attacks that target a single event (at a given time instant) in a user's trace, are called *localization* attacks. These two categories of attacks are examples of *presence/absence disclosure* attacks [SFH10]: they infer the relation between users and regions over time. In contrast, if the physical proximity between users is of interest to the adversary, we call the attack a *meeting/proximity disclosure* attack (i.e., who possibly meets whom at a given place/time).

A very general inference attack can be defined as the one that aims to recover the actual trace of each user. That is, it targets the whole set of users and the whole set of time instants, and it asks for the most likely trace of each user, or even for the whole probability distribution of traces for each user. The adversary's ultimate goal is then formally defined as calculating the following probability distribution function:

$$\Pr\{\boldsymbol{A} = \hat{\mathbf{a}} \,|\, \boldsymbol{O} = \mathbf{o}\} \tag{1.5}$$

More specific objectives can be defined, which lead to all sorts of location disclosure attacks: Specify a user and a time, and ask for the region where the user was at the specified time; specify a user and a region, and ask for the times when the user was there; specify a subset of regions, and ask for the (number of) users who visited these regions at any time.

## 1.5 Evaluation Metrics

In this section, we describe two main evaluation metrics that we formalize in our framework: location privacy of users under some inference attacks, and the service quality loss that they incur by using a LPPM to protect their privacy.

### 1.5.1 Location Privacy

At a high level, the adversary obtains some obfuscated traces $\mathbf{o}$, and, knowing the LBS, the LPPM, and the mobility profiles of the users, he tries to infer some information of interest about the actual traces $\mathbf{a}$. As we have mentioned, the possible objectives of the adversary range from the very general (the traces $\mathbf{a}$ themselves) to the specific (the location of a user at a specific time, the number of users at a particular location at a specific time, etc.).

Nevertheless, usually, neither the general nor the specific objectives have a single deterministic answer. The actual traces are generated probabilistically from the mobility profiles, the users' access patterns to the LBS are generated probabilistically according to its model (e.g., being sporadic or continuous), and the observed traces are generated probabilistically by the LPPM. So, there are many traces $\mathbf{a}$ that might have produced the observed traces $\mathbf{o}$. The same applies to the more specific objectives. For example, there are many possible regions where a user might have been at a particular time. The output of the inference attack can be a probability distribution on the possible outcomes (traces, regions, number of users),

the most probable outcome, the expected outcome under the distribution on outcomes (the average number of users), or any function of the actual trace. We call $\phi(\cdot)$ the function that describes the attacker's objective. If its argument is the actual trace $\mathbf{a}$, then its value $\phi(\mathbf{a})$ is the correct answer to the attack. Let $\mathcal{Z}$ be the set of values that $\phi(\cdot)$ can take for a given attack (e.g., $M$ regions, $N$ users, $M^T$ traces of one user, etc.).

The probabilistic nature of the attacker's task implies that he cannot obtain the exact value of $\phi(\mathbf{a})$, even if he has an infinite amount of resources. The best he can hope for is to extract all the information about $\phi(\mathbf{a})$ that is contained in the observed traces. The extracted information is in the form of the posterior distribution $\mathbb{P}\mathrm{r}\{z|\mathbf{o}\}, z \in \mathcal{Z}$, of the possible values of $\phi(\mathbf{a})$ given the observed traces $\mathbf{o}$. We call *uncertainty* the ambiguity of this posterior distribution with respect to finding a unique answer – that unique answer need not be the correct one; see the discussion on correctness later. The uncertainty is maximum, for example, if the output of a localization attack is a uniform distribution on the locations. On the contrary, the uncertainty is zero if the output is a Dirac distribution on one location.

Of course, the attacker does not have infinite resources. Consequently, the result of the attack is only an estimate $\hat{\mathbb{P}}\mathrm{r}\{z|\mathbf{o}\}$ of the posterior distribution $\mathbb{P}\mathrm{r}\{z|\mathbf{o}\}$. We call *inaccuracy* the discrepancy between the distributions $\hat{\mathbb{P}}\mathrm{r}\{z|\mathbf{o}\}$ and $\mathbb{P}\mathrm{r}\{z|\mathbf{o}\}$.

Neither the uncertainty metric nor the inaccuracy metric, however, quantify the privacy of the users. What matters for a user is whether the attacker finds the correct answer to his attack, or, alternatively, how close the attacker's output is to the correct answer. Knowing the correct answer, we can calculate a distance (or expected distortion) between the output of the attack and the true answer. We call this distance the *incorrectness* of the attack, and we believe that this is *the* appropriate way to quantify the success of an attack: The lower the expected distortion of the estimation is, the higher the correctness of the estimation is, hence the lower the privacy of users will be.

It is important that the accuracy and the certainty not be mistaken to be equivalent to the correctness of the attack. Even an attacker with infinite resources will not necessarily find the true answer, as he might have been able to observe only an insufficient number of events. But he will extract all the information that is contained in the events, so the accuracy will be maximum. If the accuracy is maximum, and the observed traces point to a unique answer – so the certainty is also maximum – the correctness will not necessarily be high. It is clearly possible, for instance, that the user did something out of the ordinary on the day the traces were collected: What she did is still consistent with the observed trace, but as it is not typical for the user, it is assigned a low probability/weight in the attack output.

### Accuracy

We compute the accuracy of each element of the distribution $\hat{\mathbb{P}}\mathrm{r}\{z|\mathbf{o}\}, z \in \mathcal{Z}$, separately. That is, we estimate the posterior probability $\mathbb{P}\mathrm{r}\{z|\mathbf{o}\}$ for each possible value $z$ of $\phi(\mathbf{a})$. We quantify the accuracy with a confidence interval and a confidence level. By definition, the probability that the accurate value of $\mathbb{P}\mathrm{r}\{z|\mathbf{o}\}$ is within the confidence interval is equal to the confidence level.

The extreme case is that the interval is of zero length (i.e., a point) and the confidence level is 1 (i.e., the attacker is absolutely confident that the point estimate is accurate). An attacker using more and more accurate estimation tools could achieve this extreme case, thus making $\hat{\mathbb{P}}\mathrm{r}\{z|\mathbf{o}\}$ converge to $\mathbb{P}\mathrm{r}\{z|\mathbf{o}\}$. However, achieving such ultimate accuracy might be prohibitively costly. So, the adversary will in general be satisfied with some high enough level

of accuracy (i.e., large enough confidence level, and small enough confidence interval). When the accuracy reaches the desired level, or the resources of the adversary are exhausted, the probability $\hat{\mathbb{P}}\mathrm{r}\{z|\mathbf{o}\}$ with some confidence interval is the estimate of the adversary.

### Certainty

We quantify the certainty with the entropy of the distribution $\hat{\mathbb{P}}\mathrm{r}\{z|\mathbf{o}\}$. The entropy shows how uniform vs. concentrated the estimated distribution is and, in consequence, how easy it is to pinpoint a single outcome $z$ out of $\mathcal{Z}$. The higher the entropy is, the lower the adversary's certainty is.

$$\hat{H}(\boldsymbol{Z}) = \sum_{z} \hat{\mathbb{P}}\mathrm{r}\{z|\mathbf{o}\} \log \frac{1}{\hat{\mathbb{P}}\mathrm{r}\{z|\mathbf{o}\}} \tag{1.6}$$

### Correctness

The *correctness* of the attack is quantified using the expected distance between the true outcome $z_c \in \mathcal{Z}$ and the estimate based on the $\hat{\mathbb{P}}\mathrm{r}\{z|\mathbf{o}\}$. In general, if there is a distortion metric $\|\cdot\|$ defined between the members of $\mathcal{Z}$, the expected distortion can be computed as the following sum, which is the adversary's *expected estimation error*:

$$\sum_{z} \hat{\mathbb{P}}\mathrm{r}\{z|\mathbf{o}\}\|z - z_c\| \tag{1.7}$$

The value $z_c$ is what the users want to hide from the adversary. The higher the adversary's correctness is, the lower the privacy of the targeted user(s) is. Hence, ***correctness is the metric that determines the privacy of users***.

The distortion metric $\|\cdot\|$ quantifies the loss of privacy stemming from the inference attack. The privacy loss depends on the locations' semantics and also on the privacy requirements of the user (e.g., a user might consider a hospital to be more sensitive than other places), and the distortion function must be defined accordingly. For the case where the adversary's objective is to find the users' locations (e.g., in tracking and localization attacks), the set $\mathcal{Z}$ becomes equivalent to the set of locations $\mathcal{R}$. In this case, we determine the value of $\|\hat{r} - r\|$ using a distortion function $d_p(\hat{r}, r)$ that is a user's privacy loss if her actual location is $r$ and the adversary estimates it as $\hat{r}$.

As an example for the distortion function, if the user wants to simply hide her *exact* current location (as opposed to hiding her location area, for example), the appropriate distortion function could be the *Hamming distortion* (probability of error), where the distance between the estimated location $\hat{r}$ and the actual location $r$ is:

$$d_p(\hat{r}, r) = \begin{cases} 0, & \text{if } \hat{r} = r \\ 1, & \text{otherwise} \end{cases} \tag{1.8}$$

In this case, any location different from the user's actual location results in a high level of location privacy. Alternatively, the user's privacy could depend on the physical distance between the estimated and actual locations, hence the distortion function can be modeled as the Euclidean distance between these locations, i.e., the *squared-error distortion*:

$$d_p(\hat{r}, r) = (\hat{r} - r)^2 \tag{1.9}$$

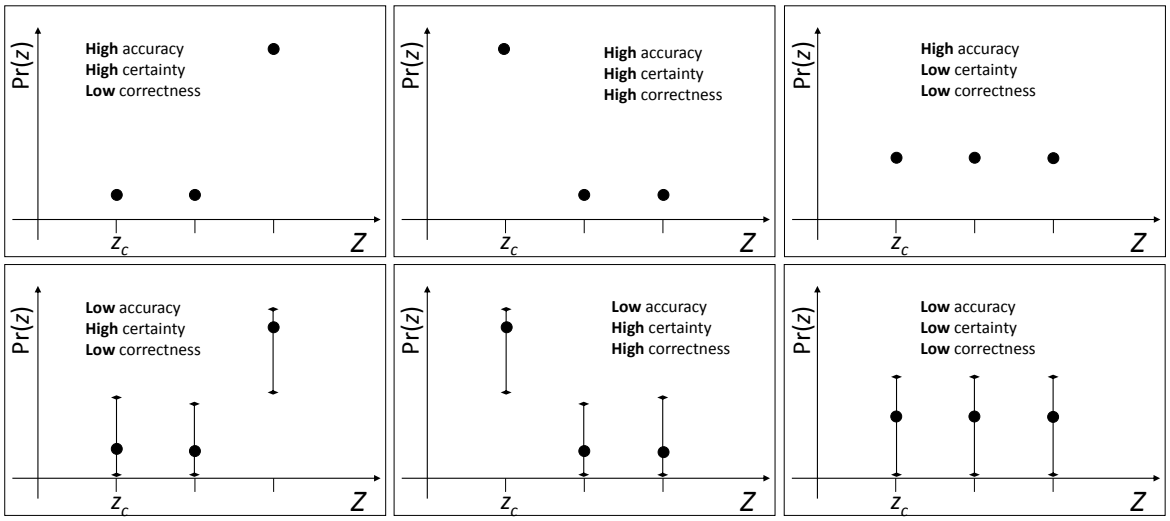



Figure 1.2: Accuracy, Certainty, and Correctness of the adversary. The adversary is estimating $\hat{\mathbb{P}}\mathrm{r}\{z|\mathbf{o}\}$ where the true value for $z$ (correct guess) is $z_c$. In this example, $z$ can get three discrete values. The black dot shows the estimate $\hat{\mathbb{P}}\mathrm{r}\{z|\mathbf{o}\}$ for different $z$ and the lines show the confidence interval for a given confidence level chosen by the adversary. As it is shown in the figures, the accuracy of the estimation is independent of its certainty and correctness. Moreover, the level of correctness does not convey anything about the level of certainty, and high certainty does not mean high correctness. The only correlation between certainty and correctness is that low certainty *usually* (depending on the size of $\mathcal{Z}$ and the distance between its members) implies low correctness.

In summary, the adversary tries to achieve the maximum accuracy for $\hat{\mathbb{P}}\mathrm{r}\{z|\mathbf{o}\}$ under his resource constraints. He can measure the success of the attack by computing the certainty over the results. However, to measure users' location privacy, given a LPPM, we consider the true value $z_c$ and measure the adversary's incorrectness (estimation error). Notice that the adversary does not know the value of $z_c$, hence he cannot evaluate this aspect of his performance. Figure 1.2 illustrates through some examples the independence of these three aspects (of the adversary's performance) from each other.

### 1.5.2 Service Quality

In the aforementioned setting, where users' actual locations are potentially altered using a LPPM, the LBS response quality depends on the pseudolocation output by the LPPM and not on the user's actual location. The distortion introduced in the observed pseudolocations determines the quality of service that each user experiences. The more similar the actual and the observed location are, the higher the service quality is.

Let function $d_q(\tilde{r}, r)$ determine the dissimilarity between location $r$ and pseudolocation $\tilde{r}$. The semantics of this dissimilarity depend on the LBS under consideration and also on the user's specific service-quality expectations. In many applications, the service quality can be considered inversely proportional to the physical distance between $r$ and $\tilde{r}$. For example, applications that find nearby points of interest could give very different responses to $r$ and to $\tilde{r}$, even if they are only a couple of kilometers apart. In contrast, there exist LBSs in which the

service quality depends on other criteria, such as on whether $\tilde{r}$ is within a region of interest. For a weather forecast application, for instance, any pseudolocation $\tilde{r}$ in the same city as the actual location $r$ would result in a high quality LBS response.

## 1.6 Summary

In this chapter, we have defined the fundamental elements of our analytical framework for location privacy. Throughout the next chapters, we further elaborate on these components and their interrelation.

**Related Publications** [SFH10, SFJH09, STD+11, STT+12, STLBH11]

# 2 User Profiling

Human beings tend to have regular life styles. This is reflected in different aspects of their everyday life, and also in the pattern at which they visit various locations. An adversarial entity, who is after discovering the true identities and location related information of LBS users, take advantage of this fact. All the information that is a priori available about each user, for example her home and workplace address, where her parents live, or her favorite points of interest, can be processed in order to construct a model for her behavior (in our case, her mobility); a model that can be used to predict the user's future activities. In this chapter, we formalize the adversary's background knowledge, which is constituted of the users' mobility profiles, as Markov chains on the set of regions (locations). Then, we use Bayesian inference techniques to estimate the transition matrix of the Markov chain associated to the mobility of each user, and also their probability distribution over the set of locations, given, e.g., the prior information about the visited locations by each user. The adversary's background knowledge will be used as the fuel for the location disclosure attacks.

## 2.1 Prior Information

The adversary collects various pieces of information about the users' mobility and the locations that they visit regularly, e.g., their homes and offices. In general, such information can be translated to events; perhaps the events can be linked into transitions; or, they can be further linked into partial traces or even full traces reflecting the trajectory of each user.

We assume, the prior mobility information exists in the following two ways: in the form of some traces $\mathbf{y}_u$, and as a matrix of transition counts $\mathbf{n}$. The traces can be noisy or they might be missing some events. The matrix $\mathbf{n}$ is of dimension $M^2$, where $M$ is the number of regions, and its $(i,j)^{\text{th}}$ entry contains the number of transitions from region $r_i$ to $r_j$, that users have had (in the past) and have not been encoded as traces. Any information about the general movement across the regions (i.e., how a typical user moves) that cannot be attributed to a particular user can be incorporated in the $\mathbf{n}$ matrix. In addition to this mobility information on the users, the adversary also considers the general *mobility constraints* of users within regions. For example, it might not be possible to move between two far-away regions in one time instant, or cross a border between two regions because of some physical obstacles. Let $\mathbf{c}$ be a binary matrix to represent this mobility constraint. The matrix $\mathbf{c}$, in some way, encodes the map of the whole area in which the users move. The summary of notations is presented in Table 2.1.

## 2.2 Background Knowledge

In this section, we first illustrate how to formalize the knowledge of the adversary. Then, we propose a method of constructing his background knowledge, given the prior information.

| | |
|---|---|
| $\mathbf{y}$ | Prior trace of the user; The element $y(t)$ is the location of user at time $t$. Trace $\mathbf{y}$ might miss some events. The trace $\hat{\mathbf{y}}$ is an estimated completion for it. |
| $\mathbf{n}$ | Prior transition counts matrix; The element $n_{r,r'}$ is the number of transitions that users have had from region $r$ to $r'$. |
| $\mathbf{c}$ | Mobility constraints matrix; The element $c_{r,r'}$ is equal to 1 if it is possible to move from $r$ to $r'$, and it is 0 otherwise. |
| $\hat{\mathbf{p}}$ | Mobility profile of the user estimated by the adversary, as the form of first-order Markov chain transition matrix; The element $\hat{p}_{r,r'}(u)$ is the probability of moving to location $r'$ given that the user $u$ has been in $r$. |
| $\hat{\boldsymbol{\pi}}$ | Mobility profile of the user estimated by the adversary, as the users' location probability distribution. The element $\hat{\pi}_r(u)$ is the probability of user $u$ visiting location $r$. |
| $\hat{\mathbf{y}}^{\{l\}}$ | The $l$th sample for trace $\hat{\mathbf{y}}$ in the Gibbs sampling procedure. |
| $\hat{\boldsymbol{\pi}}^{\{l\}}$ | The $l$th sample for location probability distribution $\hat{\mathbf{y}}$ in the Gibbs sampling procedure. |
| $\hat{\mathbf{p}}^{\{l\}}$ | The $l$th sample for transition probability matrix $\hat{\mathbf{p}}$ in the Gibbs sampling procedure. |

Table 2.1: Table of notations.

### 2.2.1 Mobility Profiles

We model user mobility as a first-order Markov chain on the set of regions (locations).[1][2] We define the mobility profile $(\mathbf{p}_u, \boldsymbol{\pi}_u)$ of a given user $u$ as the pair of a transition matrix $\mathbf{p}_u$ of the Markov chain associated with the user's mobility, and the probability distribution $\boldsymbol{\pi}_u$ of the user's location.[3] The entry $p_{r,r'}(u)$ of $\mathbf{p}_u$ is the probability that user $u$ moves to region $r'$ in the next time instant, given that she is now in region $r$. The entry $\pi_r(u)$ is the probability that user $u$ is in region $r$. Thus, the user's mobility profile consists of the following probabilities:

$$p_{r,r'}(u) = \mathbb{P}\mathrm{r}\{\boldsymbol{A}_u^{t+1} = r' \mid \boldsymbol{A}_u^t = r\}, \quad \pi_r(u) = \mathbb{P}\mathrm{r}\{\boldsymbol{A}_u^t = r\}. \tag{2.1}$$

The objective of the adversary is to estimate $\mathbf{p}_u$ and $\boldsymbol{\pi}_u$, given his prior information. Let $(\hat{\mathbf{p}}_u, \hat{\boldsymbol{\pi}}_u)$ denote the estimated values.

### 2.2.2 Profile Estimation

Hereafter, we explain how to estimate the transition probability profile $\hat{\mathbf{p}}_u$ of user $u$ from traces $\mathbf{y}_u$ with (potentially) missing events, a transition count matrix $\mathbf{n}_u$, and mobility constraint matrix $\mathbf{c}$. The stationary distribution $\hat{\boldsymbol{\pi}}_u$ can then be computed from $\hat{\mathbf{p}}_u$. Note that the method that we develop considers multiple prior traces per user. However, to simplify the presentation we consider only one trace. Moreover, as we are talking about profiling each user separately, we omit the index $u$.

[1] Note that this Markovian model can become even more precise, for example, by increasing the order of the Markov chain, or by incorporating the user activity or location semantic in each of its states.

[2] Mobility model of users is time-dependent. In this chapter, we present the user profiling algorithm considering a given time period. In the case of multiple time periods, the algorithm should be run for each time period separately.

[3] The probability distribution $\boldsymbol{\pi}_u$ is the stationary distribution of transition matrix $\mathbf{p}_u$ (if it has a stationary probability distribution).

The ultimate goal is to estimate the parameters of the underlying Markov chain that represents the user's mobility model. As the available prior trace $\mathbf{y}$ can be incomplete (i.e., location of the user might not be available for all time instants), we also need to fill in the missing data at the same time. Let $\hat{\mathbf{y}}$ be an estimated completion for $\mathbf{y}$. Formally, we estimate the profile $\hat{\mathbf{p}}$ of the user with the expectation $\mathbb{E}\{\hat{\mathbf{p}} \,|\, \mathbf{y}, \mathbf{n}, \mathbf{c}\}$. To compute this expectation we will sample from the distribution

$$\Pr\{\hat{\mathbf{p}} \,|\, \mathbf{y}, \mathbf{n}, \mathbf{c}\} = \sum_{\hat{\mathbf{y}}} \Pr\{\hat{\mathbf{p}}, \hat{\mathbf{y}} \,|\, \mathbf{y}, \mathbf{n}, \mathbf{c}\}. \tag{2.2}$$

However, sampling directly from $\Pr\{\hat{\mathbf{p}}, \hat{\mathbf{y}} \,|\, \mathbf{y}, \mathbf{n}, \mathbf{c}\}$ is not straightforward; it involves computing the sum of terms whose number grows exponentially with the length of the trace. Hence, we use Gibbs sampling algorithm [GG84, Mac03], a Markov Chain Monte Carlo (MCMC) method, as it only needs sampling from the conditional distributions $\Pr\{\hat{\mathbf{p}} \,|\, \hat{\mathbf{y}}, \mathbf{y}, \mathbf{n}, \mathbf{c}\}$ and $\Pr\{\hat{\mathbf{y}} \,|\, \hat{\mathbf{p}}, \mathbf{y}, \mathbf{n}, \mathbf{c}\}$. More precisely, in order to sample from $\Pr\{\hat{\mathbf{p}}, \hat{\mathbf{y}} \,|\, \mathbf{y}, \mathbf{n}, \mathbf{c}\}$, we create a homogeneous Markov chain on the state space of $\hat{\mathbf{p}}$ and $\hat{\mathbf{y}}$ in an iterative procedure. Starting from an initial value for $\hat{\mathbf{y}}^{\{0\}}$, Gibbs sampling produces pairs $(\hat{\mathbf{p}}^{\{l\}}, \hat{\mathbf{y}}^{\{l\}})$ as follows:

$$\hat{\mathbf{p}}^{\{l\}} \sim \Pr\{\hat{\mathbf{p}} \,|\, \hat{\mathbf{y}}^{\{l-1\}}, \mathbf{y}, \mathbf{n}, \mathbf{c}\} \tag{2.3}$$

$$\hat{\mathbf{y}}^{\{l\}} \sim \Pr\{\hat{\mathbf{y}} \,|\, \hat{\mathbf{p}}^{\{l\}}, \mathbf{y}, \mathbf{n}, \mathbf{c}\} \tag{2.4}$$

As we discuss later, the sequence of the $\hat{\mathbf{p}}^{\{l\}}$ matrices converges to the desired probability distribution $\Pr\{\hat{\mathbf{p}} \,|\, \mathbf{y}, \mathbf{n}, \mathbf{c}\}$, and we estimate the user's profile by computing the average value of the samples. We then compute $\hat{\boldsymbol{\pi}}$ as the steady state distribution of $\hat{\mathbf{p}}$. Now, let us explain how to produce the samples.

**Sampling** (2.3)

In order to sample a $\hat{\mathbf{p}}^{\{l\}}$ from (2.3), according to the Markov property, we assume that the rows of the transition matrix $\hat{\mathbf{p}}$ are independent. More precisely, given that the user is in region $r$, the probability that she moves to a region in $\mathcal{R}$ is independent of her transition probability from any region $r'$ to $r'' \in \mathcal{R}$ when $r' \neq r$. So, we produce samples for each row of the matrix separately. We consider a Dirichlet prior for each row of $\hat{\mathbf{p}}$, based on the prior observation on the number of transitions from a region $r$ to other regions. Dirichlet distribution is the conjugate prior for multinomial and categorical distributions, where the outcome of a random experiment is one of $M$ rival events (in our case, going to any of $M$ regions from a given region $r$). That is, its probability density function reflects the belief about the probabilities of $M$ rival events, given the number of observations about each event. Hence, we generate the elements of the $l^{\text{th}}$ sample for $\hat{\mathbf{p}}$ by sampling according to the following Dirichlet distribution:

$$\hat{\mathbf{p}}_{r,\cdot}^{\{l\}} \sim \mathrm{Dir}\big(\{n_{r,r'} + \mathsf{cnt}_{r,r'}(\hat{\mathbf{y}}^{\{l-1\}}) + \epsilon c_{r,r'}\}_{\forall r' \in \mathcal{R}}\big), \forall r \in \mathcal{R} \tag{2.5}$$

where $\mathsf{cnt}_{r,r'}(\cdot)$ is the number of transitions from region $r$ to $r'$ in a trace, $\epsilon$ is a very small positive number, and $c_{r,r'}$ is an element of the mobility constraint binary matrix $\mathbf{c}$ corresponding to the possibility of moving from region $r$ to $r'$ in one time instant. The term $\epsilon c_{r,r'}$ (if $c_{r,r'} = 1$) gives some small positive weight to the transition between two regions $r$ and $r'$, even if there is no transition between them in the adversary's prior information. The rationale is that the absence of prior observation on transition between $r$ and $r'$ (in the finite trace $\mathbf{y}$, or in matrix $\mathbf{n}$) does not mean that the probability of moving between $r$ and $r'$ is zero.

**Sampling** (2.4)

To sample a complete trace $\hat{\mathbf{y}}^{\{l\}}$ from (2.4), we follow the procedure proposed in [RCD93] and iteratively construct $\hat{\mathbf{y}}^{\{l\}}$ by performing $T$ successive samplings,[4] from

$$\frac{\hat{p}^{\{l\}}_{\hat{y}(t-1),\hat{y}(t)} \cdot \mathsf{o}(y(t) \,|\, \hat{y}(t)) \cdot \hat{p}^{\{l\}}_{\hat{y}(t),\hat{y}(t+1)}}{\sum_r \hat{p}^{\{l\}}_{\hat{y}(t-1),r} \cdot \mathsf{o}(y(t) \,|\, r) \cdot \hat{p}^{\{l\}}_{r,\hat{y}(t+1)}}, \quad t = 1, \ldots, T, \tag{2.6}$$

where $\hat{p}_{\hat{y}(t),\hat{y}(t+1)}$ is, according to $\hat{\mathbf{p}}^{\{l\}}$, the probability of moving from the region visited at time $t$, in the trace $\hat{\mathbf{y}}$, to its subsequent visited location at time $t+1$. The value of the probability $\hat{p}^{\{l\}}_{\hat{y}(t),\hat{y}(t+1)}$ for $t = 0$ and $t = T$ is defined to be 1. The function $\mathsf{o}(r'|r)$ represents the noise function if the trace $\mathbf{y}$ is noisy: It is the probability that user location $r$ is distorted as $r'$ in the trace (for example due to positioning error). If $r'$ is missing (due to the incompleteness of the trace), then $\mathsf{o}(r'|r)$ is equal to 1.

**Estimating the Profile given the Samples**

The convergence properties of the Gibbs sampling for this problem are studied in [RCD93]. As it is shown in [DR92, Gey91, RCD93, Tie91], the illustrated Gibbs sampling, i.e., the iterative simulation of $\hat{\mathbf{p}}^{\{l\}}$ according to (2.5) and $\hat{\mathbf{y}}^{\{l\}}$ according to (2.6), produces a homogeneous, irreducible and aperiodic Markov chain $(\hat{\mathbf{p}}^{\{l\}}, \hat{\mathbf{y}}^{\{l\}})$ that converges at a geometric rate to a stationary distribution.[5][6] Hence, the sequence of the $\hat{\mathbf{p}}^{\{l\}}$ matrices is ergodic and converges to the desired probability distribution $\Pr\{\hat{\mathbf{p}} \,|\, \mathbf{y}, \mathbf{n}, \mathbf{c}\}$. Thus, the Ergodic theorem for Markov chains applies, and we can compute $\hat{p}_{r,r'}$ for every $r, r'$ as the average of $\hat{p}^{\{l\}}_{r,r'}$ over all samples $l$, for a large enough $L$ number of samples:

$$\hat{p}_{r,r'} = \frac{1}{L} \sum_l \hat{p}^{\{l\}}_{r,r'}, \; \forall r, r' \in \mathcal{R}. \tag{2.7}$$

The higher the number of iterations is, the higher the accuracy of the user mobility profile will be. The sample variance of each element of the mobility transition matrix is a candidate for the accuracy metric. However, if the learning trace is complete (i.e., it does not miss any location event) and not noisy, then the transition matrix samples in each iteration construct an i.i.d. sequence, and consequently we can use the variance of each element and the number of samples to compute a confidence interval on each element of the mobility matrix. This can be done, for example, using Theorem 2.2 of [LB10] where the confidence interval at level $\gamma$ for the sample mean $\hat{p}_{r,r'}$ is $\hat{p}_{r,r'} \pm \frac{1+\gamma}{2\sqrt{L}} \hat{v}_{r,r'}$ where $\hat{v}_{r,r'}$ is the sample variance of $\hat{p}^{\{l\}}_{r,r'}$.

[4] Without loss of generality, we assume that trace $\mathbf{y}$ spans from time $t = 1$ to $T$.

[5] A Markov chain ($X(t)$ being the state at time $t$) is called time-homogeneous (or just *homogeneous*) if $\Pr\{X(t+k) = j \,|\, X(t) = i\}$ is independent of $t$. It is called *irreducible*, if it is possible to get to any state from any state, i.e., $\forall i, j, \exists k$, such that $\Pr\{X(t+k) = j \,|\, X(t) = i\} > 0$. It is called *aperiodic* if all the states are aperiodic, i.e., for any state $i$, $\mathsf{gcd}\{k \geq 1 : \Pr\{X(t+k) = i \,|\, X(t) = i\} > 0\} = 1$, where function $\mathsf{gcd}$ denotes the greatest common divisor [KT75].

[6] The Ergodic theorem simply states that if a sequence $\{X(0), X(1), ..., X(n)\}$ is an irreducible, homogeneous discrete Markov chain with some stationary distribution $\pi$, then the sample mean of $f(X(i))$, for some bounded function $f : \mathfrak{X} \longrightarrow \mathbb{R}$, converges to the true distribution of $\mathbb{E}\{f(X)\}$. If further, it is aperiodic, then the probability $\Pr\{X(n) = x \,|\, X(0) = x_0\}$ converges to $\pi(x)$ for any $x, x_0 \in \mathfrak{X}$.

## 2.3 Summary

In this chapter, we have used Gibbs sampling, which is a Markov Chain Monte Carlo method, in order to construct the mobility profile of users from some prior information about their mobility. This mobility profile will be used by the adversary in his inference attacks in order to invert the location-privacy preserving mechanisms and disclose private information about the users' actual locations.

**Related Publications** [STLBH11]

# 3 Inference Attacks

Our goal, in this chapter, is to make progress on the **quantification** of the performance of a location-privacy preserving mechanism (LPPM). This is an important topic, because (i) people are notoriously bad estimators of risks in general, and privacy risks in particular, (ii) it is the only way to make meaningful comparisons between different LPPMs and (iii) the research literature is not yet mature enough on the topic.

In specific areas, several contributions have been made to quantify privacy, be it for databases [Dwo06], for anonymity protocols [CPP08], or for anonymization networks [DSCP02, SD02, Tro11, TD09]. Yet, in the field of location privacy, notwithstanding many contributions from different disciplines (such as databases, mobile networks, and ubiquitous computing [Kru09b]) for protecting location privacy, the lack of a unified and generic formal framework for specifying protection mechanisms and also for evaluating location privacy is evident. This has led to the divergence of contributions and, hence, has caused confusion about which mechanisms are more effective. In general, the adversary model is often not appropriately addressed and formalized, and a good model for adversary's possible inference attacks (while incorporating his background knowledge) is missing. This can lead to a wrong estimation of the location privacy of mobile users. In few cases, that privacy vulnerabilities of sharing location data is discussed, location privacy is evaluated as the anonymity of users [BJB+12, MYYR10, DMDBP08, ZB11]. Hence, the amount of location information leakage through sharing location data is not properly quantified. Moreover, the proposed evaluation schemes cannot be extended to generic location-privacy preserving mechanisms.

In this chapter, we propose a generic theoretical framework for quantifying location privacy of LBS users. We make the following contributions.

- We provide a generic model that formalizes the adversary's attacks against private location-information of mobile users. In particular, we rigorously define *de-anonymization* (re-identifying pseudonymous location traces), *localization* (finding location of a user at a specific time) and *tracking* attacks on anonymous traces as statistical inference problems. We also construct a generic attack that discloses even more location-related information about users who expose their (pseudonymous and obfuscated) locations to LBSs.

- We rely on well-established statistical methods to evaluate the performance of such inference attacks. More precisely, we rely on Bayesian inference techniques, especially those that are proposed for hidden Markov models. We infer users' private information even if their LBS accesses are pseudonymous[1] and obfuscated. We formalize the adversary's success and we clarify, explain and justify the right metric to quantify location privacy: The adversary's *expected estimation error*.

[1] In this chapter, we assume persistent pseudonyms for users.

- We provide a tool: the *Location-Privacy and Mobility Meter* (LPM) is developed based on our formal framework and is designed for evaluating the effectiveness of various location-privacy preserving mechanisms. We use LPM to evaluate various location-privacy preserving mechanisms and to compare their effectiveness under similar settings.
- We show the inappropriateness of some existing metrics, notably entropy and k-anonymity, for quantifying location privacy.

## 3.1 De-Anonymization Attack

The de-anonymization attack finds the most likely assignment of users $\{u_1, u_2, ..., u_N\}$ to the observed (pseudonymous and obfuscated) traces $\{\mathbf{o}_{\tilde{u}_1}, \mathbf{o}_{\tilde{u}_2}, ..., \mathbf{o}_{\tilde{u}_N}\}$. Notice that it is not correct to simply assign each user to the trace that she could have most likely created. This is because doing so more than one user might be assigned to the same trace. The most likely assignment is a *joint* assignment; it maximizes the joint probability of assigning all users to the observed traces (pseudonyms). More formally, we compute the following:

$$\boldsymbol{\sigma}^* = \arg\max_{\sigma} \Pr\{\boldsymbol{\Sigma} = \boldsymbol{\sigma} \,|\, \boldsymbol{O} = \mathbf{o}\}. \tag{3.1}$$

### 3.1.1 User-Pseudonym Likelihood Graph

We use Bayesian inference in order to perform the de-anonymization. In order to obtain the most likely assignment $\boldsymbol{\sigma}^*$, we need to maximize the following probability:

$$\Pr\{\boldsymbol{\Sigma} = \boldsymbol{\sigma} \,|\, \boldsymbol{O} = \mathbf{o}\} = \frac{\Pr\{\mathbf{o} \,|\, \boldsymbol{\sigma}\} \cdot \Pr\{\boldsymbol{\sigma}\}}{\Pr\{\mathbf{o}\}} = \prod_{\tilde{u}} \Pr\{\mathbf{o}_{\tilde{u}} \,|\, \boldsymbol{\sigma}\} \cdot \underbrace{\frac{\Pr\{\boldsymbol{\sigma}\} \equiv \frac{1}{N!}}{\Pr\{\mathbf{o}\}}}_{\text{constant}}. \tag{3.2}$$

Thus,

$$\boldsymbol{\sigma}^* = \arg\max_{\boldsymbol{\sigma}} \prod_{\tilde{u}} \Pr\{\mathbf{o}_{\tilde{u}} \,|\, \boldsymbol{\sigma}\} \tag{3.3}$$

Notice that, given the assignment of a user $u$ to a pseudonym $\tilde{u}$, the probability $\Pr\{\mathbf{o}_{\tilde{u}} \,|\, \boldsymbol{\sigma}\}$ is independent of all other user-pseudonym assignments. So, to find the most likely assignment $\boldsymbol{\sigma}^*$, we first compute $\Pr\{\mathbf{o}_{\tilde{u}} \,|\, \sigma(u) = \tilde{u}\}$ for all pairs of $u \in \mathcal{U}$ and $\tilde{u} \in \widetilde{\mathcal{U}}$. Then, we construct a complete weighted bipartite graph in which disjoint sets of vertices are $\mathcal{U}$ and $\widetilde{\mathcal{U}}$, and the weight on the edge between given vertices $u$ and $\tilde{u}$ is the likelihood $\Pr\{\mathbf{o}_{\tilde{u}} \,|\, \sigma(u) = \tilde{u}\}$. We later use this graph to de-anonymize the users.

For the case when the obfuscation function operates on each region separately, we compute the likelihood for each pair as follows. These probabilities should be computed appropriately according to the background knowledge that we consider for the adversary. In the next subsections, we compute these probabilities for two adversaries with different background knowledge:

**I** The adversary whose knowledge of users' mobility is the probability distribution over the regions for each user, i.e., $\hat{\boldsymbol{\pi}}$.

**II** The adversary who, in addition to knowing $\hat{\boldsymbol{\pi}}$, also knows the users' probability of transition between the regions, i.e., $\hat{\mathbf{p}}$.

| | |
|---|---|
| $\mathsf{LBS}(\theta)$ | A location-based service with users' access probability $\theta$. |
| $\mathsf{LPPM}(\mathsf{A}, \mathsf{O}\rho, \mathsf{F}\phi\{\mathsf{u}\|\mathsf{g}\})$ | A location-privacy preserving mechanism that ($\mathsf{A}$) anonymizes traces, ($\mathsf{O}\rho$) obfuscates each location among $2^\rho$ locations, and ($\mathsf{F}\phi\{\mathsf{u}\|\mathsf{g}\}$) accesses the $\mathsf{LBS}$ on behalf of the users and generates fake locations: ($\mathsf{u}$) uniformly from the set of locations, or ($\mathsf{g}$) according to the average users' location distribution $\bar{\pi}(r) = \frac{1}{N}\sum_u \hat{\boldsymbol{\pi}}_u(r)$. |
| $\boldsymbol{\sigma}$ | An assignment of users to pseudonyms. Element $\sigma(u)$ is the pseudonym assigned to user $u$. |
| $\boldsymbol{\sigma}^*$ | The most likely assignment of users to pseudonyms (observed traces) given the adversary knowledge and the observed traces. |
| $\alpha_r^{u,\tilde{u}}(t)$ | The forward variable (in the forward-backward algorithm), which is the joint probability of observed trace $\mathbf{o}_{\tilde{u}}$ up to time $t$, and that the actual location of user $u$ at time $t$ is $r$, given that pseudonym $\tilde{u}$ is associated with user $u$. |
| $\beta_r^{u,\tilde{u}}(t)$ | The backward variable (in the forward-backward algorithm), which is the probability of observed trace $\mathbf{o}_{\tilde{u}}$ from time $t+1$ to the end, given that the actual location of user $u$ at time $t$ is $r$ and given that the pseudonym $\tilde{u}$ is associated with user $u$. |
| $\hat{\mathbf{a}}_u^*$ | The most likely trace of user $u$, given the adversary knowledge on the user's mobility profile, and given an observed trace associated to $u$. |
| $\delta_r^u(t)$ | The recursive variable used in the Viterbi algorithm, which is the joint probability of the most likely trace of user $u$ up to time $t-1$, and that her location at time $t$ is $r$, and that her observed trace up to the time $t-1$, given the adversary knowledge on her mobility profile. |
| $LP(u,t)$ | Location privacy of user $u$ at time $t$, against the localization attack, quantified with the distortion metric (incorrectness or expected estimation error of the adversary). |
| $NK(u,t)$ | Location privacy of user $u$ at time $t$, as her normalized k-anonymity. |
| $NH(u,t)$ | Location privacy of user $u$ at time $t$, against the localization attack, quantified with the normalized entropy metric. |

Table 3.1: Table of notations.

### 3.1.2 Likelihood Computation Given the Mobility Profile as $\hat{\boldsymbol{\pi}}$

For adversary (I), whose knowledge is limited to $\hat{\boldsymbol{\pi}}$, we compute the likelihood of assigning observed trace $\mathbf{o}_{\tilde{u}}$ to user $u$ as follows:

$$\mathbb{P}\mathrm{r}\{\mathbf{o}_{\tilde{u}} \,|\, \sigma(u) = \tilde{u}, \hat{\boldsymbol{\pi}}\} = \prod_{t\in\mathcal{T}} \sum_{r\in\mathcal{R}} \sum_{x\in\{0,1\}} \underbrace{\mathbb{P}\mathrm{r}\{o_{\tilde{u}}^t \,|\, \boldsymbol{X}_u^t = x, \boldsymbol{A}_u^t = r, \sigma(u) = \tilde{u}, \hat{\boldsymbol{\pi}}\}}_{\text{LPPM - Obfuscation mechanism}} \cdot \underbrace{\mathbb{P}\mathrm{r}\{\boldsymbol{X}_u^t = x \,|\, \boldsymbol{A}_u^t = r, \hat{\boldsymbol{\pi}}\}}_{\text{LBS Application}} \cdot \underbrace{\mathbb{P}\mathrm{r}\{\boldsymbol{A}_u^t = r \,|\, \hat{\boldsymbol{\pi}}\}}_{\text{User Mobility Profile}} \tag{3.4}$$

### 3.1.3 Likelihood Computation Given the Mobility Profile as $(\hat{\boldsymbol{\pi}}, \hat{\mathbf{p}})$

For adversary (II), whose knowledge expands beyond $\hat{\boldsymbol{\pi}}$ and who also knows $\hat{\mathbf{p}}$, we compute the likelihood for each pair of observed trace $\mathbf{o}_{\tilde{u}}$ and user $u$ with the *Forward-Backward* algorithm [Rab89]. This is an inference algorithm for hidden Markov models; it computes the posterior marginals of all hidden state variables given a sequence of observations. With this algorithm, each likelihood computation takes time $O(TM^2)$ by taking advantage of the recursive nature of the likelihood that we want to compute.

We first define the *forward* variable $\alpha_r^{u,u'}(t)$ for $t \in \mathcal{T}, r \in \mathcal{R}, u \in \mathcal{U}, \tilde{u} \in \widetilde{\mathcal{U}}$ as

$$\alpha_r^{u,\tilde{u}}(t) = \mathbb{P}\mathrm{r}\{\boldsymbol{A}_u^t = r, \mathbf{o}_{\tilde{u}}^{1:t} \,|\, \sigma(u) = \tilde{u}, \hat{\mathbf{p}}\}, \tag{3.5}$$

which is the joint probability of (i) the observed trace $\mathbf{o}_{\tilde{u}}$ up to time $t$ and (ii) that the actual location of the user with pseudonym $\tilde{u}$ is $r$ at time $t$, given that the pseudonym $\tilde{u}$ is associated with user $u$. Notice that, if we can compute the forward variable at all regions at time $T$, i.e., $\alpha_r^{u,u'}(T)$, then the desired likelihood is simply

$$\mathbb{P}\mathrm{r}\{\mathbf{o}_{\tilde{u}} \,|\, \sigma(u) = \tilde{u}, \hat{\mathbf{p}}\} = \sum_{r \in \mathcal{R}} \mathbb{P}\mathrm{r}\{\boldsymbol{A}_u^T = r, \mathbf{o}_{\tilde{u}} \,|\, \sigma(u) = \tilde{u}, \hat{\mathbf{p}}\} = \sum_{r \in \mathcal{R}} \alpha_r^{u,\tilde{u}}(T). \tag{3.6}$$

We compute the forward variables, for $1 \leq t < T$, recursively as

$$\begin{aligned}
\alpha_r^{u,\tilde{u}}(t+1) &= \mathbb{P}\mathrm{r}\{\boldsymbol{A}_u^{t+1} = r, \mathbf{o}_{\tilde{u}}^{1:t+1} \,|\, \sigma(u) = \tilde{u}, \hat{\mathbf{p}}\} \\
&= \sum_{x \in \{0,1\}} \underbrace{\mathbb{P}\mathrm{r}\{o_{\tilde{u}}^{t+1} \,|\, \boldsymbol{X}_u^{t+1} = x, \boldsymbol{A}_u^{t+1} = r, \sigma(u) = \tilde{u}, \hat{\mathbf{p}}\}}_{\text{LPPM - Obfuscation mechanism}} \cdot \underbrace{\mathbb{P}\mathrm{r}\{\boldsymbol{X}_u^{t+1} = x \,|\, \boldsymbol{A}_u^{t+1} = r, \hat{\mathbf{p}}\}}_{\text{LBS Application}} \\
&\quad \cdot \sum_{r' \in \mathcal{R}} \underbrace{\mathbb{P}\mathrm{r}\{\boldsymbol{A}_u^{t+1} = r \,|\, \boldsymbol{A}_u^t = r', \hat{\mathbf{p}}\}}_{\text{User Mobility Profile}} \cdot \underbrace{\mathbb{P}\mathrm{r}\{\boldsymbol{A}_u^t = r', \mathbf{o}_{\tilde{u}}^{1:t} \,|\, \sigma(u) = \tilde{u}, \hat{\mathbf{p}}\}}_{= \alpha_{r'}^{u,\tilde{u}}(t)}.
\end{aligned} \tag{3.7}$$

Within the inner sum, there is one term for each way of reaching region $r$ at time $t+1$, i.e., having been at each of the regions $r' \in \mathcal{R}$ at time $t$. After computing the sum, we need only to multiply with the probability of sharing location $r$ through the LBS application and then obfuscating it to the pseudolocation observed at time $t+1$, summed over two possibilities of sharing or not sharing the location. Now, the only remaining issue is the initialization of the forward variables:

$$\begin{aligned}
\alpha_r^{u,\tilde{u}}(1) &= \mathbb{P}\mathrm{r}\{\boldsymbol{A}_u^1 = r, o_{\tilde{u}}^1 \,|\, \sigma(u) = \tilde{u}, \hat{\mathbf{p}}\} \\
&= \sum_{x \in \{0,1\}} \underbrace{\mathbb{P}\mathrm{r}\{o_{\tilde{u}}^1 \,|\, \boldsymbol{A}_u^1 = r, \boldsymbol{X}_u^1 = x, \sigma(u) = \tilde{u}, \hat{\mathbf{p}}\}}_{\text{LPPM - Obfuscation mechanism}} \cdot \underbrace{\mathbb{P}\mathrm{r}\{\boldsymbol{X}_u^1 = x \,|\, \boldsymbol{A}_u^1 = r, \hat{\mathbf{p}}\}}_{\text{LBS Application}} \\
&\quad \cdot \underbrace{\mathbb{P}\mathrm{r}\{\boldsymbol{A}_u^1 = r \,|\, \hat{\mathbf{p}}\}}_{\text{User Mobility Profile}}.
\end{aligned} \tag{3.8}$$

Hence, we first compute (3.8) for all locations, and then iteratively for $t = 1..T$ we compute the value of forward variables for all locations and time instants using (3.7). Having computed $\alpha_r^{u,\tilde{u}}(T), \forall r \in \mathcal{R}$, we finally compute the desired likelihood using (3.6). The whole likelihood computation for one pair of observed trace and user can be done in $2M(T(M+2)-(M+1))$ multiplications and $2M(M(T-1)+1)$ additions.

For the computation of the likelihood, we do not need the backward variables (which is where the rest of the algorithm's name comes from). We will, however, define and use them in Section 3.2 on localization attacks.

### 3.1.4 Maximum Weight Assignment

Having computed the likelihoods for all pairs of observed traces and users in the bipartite graph, we complete the de-anonymization attack by solving the *Maximum Weight Assignment* (MWA) problem in this graph[2], in order to obtain $\boldsymbol{\sigma}^*$. We use the Hungarian algorithm, which has time complexity of order $O(N^4)$. Faster algorithms exist, but the Hungarian algorithm is simple, and it only needs to be computed once in this attack. The MWA is also an instance of a linear program, so linear program solvers can be used. The outcome is a matching of users and observed traces, such that the product $\prod_{\tilde{u}} \Pr\{\mathbf{o}_{\tilde{u}} \,|\, \sigma(u) = \tilde{u}\}$ is maximized over all $N!$ user-to-trace assignments.

## 3.2 Localization Attack

Localization is the attack in which the adversary computes the probability distribution over regions where a specific user might be at a specific time instant. This probability is computed given the observed traces from users. More formally, the adversary computes $\Pr\{\boldsymbol{A}_u^t = r \,|\, \mathbf{o}\}$ for user $u$ at time instant $t$ for all regions $r \in \mathcal{R}$.

Then, given the most likely assignment of users to observed traces $\boldsymbol{\sigma}^*$, which we computed in (3.1) through the de-anonymization attack, we compute the probability distribution of the given user's location at the given time instant as $\Pr\{\boldsymbol{A}_u^t = r \,|\, \mathbf{o}, \boldsymbol{\Sigma} = \boldsymbol{\sigma}^*\}$, which itself can be computed as $\Pr\{\boldsymbol{A}_u^t = r \,|\, \mathbf{o}_{\tilde{u}}, \sigma^*(u) = \tilde{u}\}$.

### 3.2.1 Localization Given the Mobility Profile as $\hat{\boldsymbol{\pi}}$

Depending on the adversary knowledge, the probability of interest for localization is computed differently. In the case when the adversary knows only the probability distribution over locations $\hat{\boldsymbol{\pi}}$, the probability that user $u$ at time $t$ is in region $r$ is

$$\begin{aligned}\Pr\{\boldsymbol{A}_u^t = r \,|\, \mathbf{o}_{\tilde{u}}, \sigma^*(u) = \tilde{u}, \hat{\boldsymbol{\pi}}\} &= \frac{\Pr\{\boldsymbol{A}_u^t = r, o_{\tilde{u}}^t \,|\, \mathbf{o}_{\tilde{u}}^{1:t-1,t+1:T}, \sigma^*(u) = \tilde{u}, \hat{\boldsymbol{\pi}}\}}{\Pr\{\mathbf{o}_{\tilde{u}} \,|\, \sigma^*(u) = \tilde{u}, \hat{\boldsymbol{\pi}}\}} \\ &= \frac{\Pr\{\boldsymbol{A}_u^t = r, o_{\tilde{u}}^t \,|\, \sigma^*(u) = \tilde{u}, \hat{\boldsymbol{\pi}}\}}{\Pr\{\mathbf{o}_{\tilde{u}} \,|\, \sigma^*(u) = \tilde{u}, \hat{\boldsymbol{\pi}}\}}\end{aligned} \tag{3.9}$$

where the normalizing factor $\Pr\{\mathbf{o}_{\tilde{u}} \,|\, \sigma^*(u) = \tilde{u}, \hat{\boldsymbol{\pi}}\}$ is $\sum_{r' \in \mathcal{R}} \Pr\{\boldsymbol{A}_u^t = r', o_{\tilde{u}}^t \,|\, \sigma^*(u) = \tilde{u}, \hat{\boldsymbol{\pi}}\}$, and the numerator can be computed as follows:

$$\begin{aligned}\Pr\{\boldsymbol{A}_u^t = r, o_{\tilde{u}}^t \,|\, \sigma^*(u) = \tilde{u}, \hat{\boldsymbol{\pi}}\} &= \Pr\{o_{\tilde{u}}^t \,|\, \boldsymbol{A}_u^t = r, \sigma^*(u) = \tilde{u}, \hat{\boldsymbol{\pi}}\} \cdot \Pr\{\boldsymbol{A}_u^t = r \,|\, \hat{\boldsymbol{\pi}}\} \\ &= \sum_{x \in \{0,1\}} \underbrace{\Pr\{o_{\tilde{u}}^t \,|\, \boldsymbol{X}_u^t = x, \boldsymbol{A}_u^t = r, \sigma^*(u) = \tilde{u}, \hat{\boldsymbol{\pi}}\}}_{\text{LPPM - Obfuscation mechanism}} \\ &\qquad \cdot \underbrace{\Pr\{\boldsymbol{X}_u^t = x \,|\, \boldsymbol{A}_u^t = r, \hat{\boldsymbol{\pi}}\}}_{\text{LBS Application}} \\ &\qquad \cdot \underbrace{\Pr\{\boldsymbol{A}_u^t = r \,|\, \hat{\boldsymbol{\pi}}\}}_{\text{User Mobility Profile}}\end{aligned} \tag{3.10}$$

[2]See also [TGPV08] for another example of using MWA in user de-anonymization attacks.

### 3.2.2 Localization Given the Mobility Profile as $(\hat{\boldsymbol{\pi}}, \hat{\mathbf{p}})$

For the stronger adversary, who knows $\hat{\boldsymbol{\pi}}$ and $\hat{\mathbf{p}}$, we use the Bayesian rule to compute the localization probability

$$\Pr\{\boldsymbol{A}_u^t = r \,|\, \mathbf{o}_{\tilde{u}}, \sigma^*(u) = \tilde{u}, \hat{\mathbf{p}}\} = \frac{\Pr\{\boldsymbol{A}_u^t = r, \mathbf{o}_{\tilde{u}} \,|\, \sigma^*(u) = \tilde{u}, \hat{\mathbf{p}}\}}{\Pr\{\mathbf{o}_{\tilde{u}} \,|\, \sigma^*(u) = \tilde{u}, \hat{\mathbf{p}}\}}, \tag{3.11}$$

where the numerator probability can be easily computed with the Forward-Backward algorithm. In Section 3.1, we described the computation of the forward variables

$$\alpha_r^{u,\tilde{u}}(t) = \Pr\{\boldsymbol{A}_u^t = r, \mathbf{o}_{\tilde{u}}^{1:t} \,|\, \sigma(u) = \tilde{u}, \hat{\mathbf{p}}\}. \tag{3.12}$$

The backward variables are defined to be

$$\beta_r^{u,\tilde{u}}(t) = \Pr\{\mathbf{o}_{\tilde{u}}^{t+1:T} \,|\, \boldsymbol{A}_u^t = r, \sigma(u) = \tilde{u}, \hat{\mathbf{p}}\}, \tag{3.13}$$

that is, $\beta_r^{u,\tilde{u}}(t)$ is the probability of the partial observed trace from time $t+1$ to the end, given that the trace $\mathbf{o}_{\tilde{u}}$ is observed from user $u$ whose actual location at time $t$ is $r$.

The computations of backward variables (3.13), for $t < T$, are done recursively as follows:

$$\begin{aligned}
\beta_r^{u,\tilde{u}}(t) &= \Pr\{\mathbf{o}_{\tilde{u}}^{t+1:T} \,|\, \boldsymbol{A}_u^t = r, \sigma(u) = \tilde{u}, \hat{\mathbf{p}}\} \\
&= \sum_{r' \in \mathcal{R}} \underbrace{\Pr\{\mathbf{o}_{\tilde{u}}^{t+2:T} \,|\, \boldsymbol{A}_u^{t+1} = r', \sigma(u) = \tilde{u}, \hat{\mathbf{p}}\}}_{= \beta_{r'}^{u,\tilde{u}}(t+1)} \\
&\quad \cdot \sum_{x \in \{0,1\}} \underbrace{\Pr\{o_{\tilde{u}}^{t+1} \,|\, \boldsymbol{X}_u^{t+1} = x, \boldsymbol{A}_u^{t+1} = r', \sigma(u) = \tilde{u}, \hat{\mathbf{p}}\}}_{\text{LPPM - Obfuscation mechanism}} \\
&\quad \cdot \underbrace{\Pr\{\boldsymbol{X}_u^{t+1} = x \,|\, \boldsymbol{A}_u^{t+1} = r', \hat{\mathbf{p}}\}}_{\text{LBS Application}} \\
&\quad \cdot \underbrace{\Pr\{\boldsymbol{A}_u^{t+1} = r' \,|\, \boldsymbol{A}_u^t = r, \hat{\mathbf{p}}\}}_{\text{User Mobility Profile}}
\end{aligned} \tag{3.14}$$

Note that the computation takes place backwards in time. The initialization (at time $T$) of the backward variables is:

$$\beta_r^{u,\tilde{u}}(T) = 1, \ \forall r \in \mathcal{R}, u \in \mathcal{U}, \tilde{u} \in \widetilde{\mathcal{U}}. \tag{3.15}$$

Having computed the backward variables, the probability $\Pr\{\boldsymbol{A}_u^t = r \,|\, \mathbf{o}_{\tilde{u}}, \sigma^*(u) = \tilde{u}\}$ is then equal to

$$\Pr\{\boldsymbol{A}_u^t = r \,|\, \mathbf{o}_{\tilde{u}}, \sigma^*(u) = \tilde{u}, \hat{\mathbf{p}}\} = \frac{\alpha_r^{u,\tilde{u}}(t) \cdot \beta_r^{u,\tilde{u}}(t)}{\Pr\{\mathbf{o}_{\tilde{u}} \,|\, \sigma^*(u) = \tilde{u}, \hat{\mathbf{p}}\}}. \tag{3.16}$$

The variable $\alpha_r^{u,\tilde{u}}(t)$ accounts for the observations up to time $t$ and region $r$ at time $t$, and $\beta_r^{u,\tilde{u}}(t)$ accounts for the remainder of the observed trace, given that the user's location at time $t$ is region $r$. The term $\Pr\{\mathbf{o}_{\tilde{u}} \,|\, \sigma^*(u) = \tilde{u}, \hat{\mathbf{p}}\}$ is a normalization factor that was earlier (3.6) computed as $\sum_{r \in \mathcal{R}} \alpha_r^{u,\tilde{u}}(T)$. An alternative way of computing it is as $\sum_{r \in \mathcal{R}} \alpha_r^{u,\tilde{u}}(t) \cdot \beta_r^{u,\tilde{u}}(t)$, which more directly shows its role as a normalization factor.

### 3.2.3 Meeting Disclosure Attack

The results of the localization attack can be used to infer even more location-related information about the users. For an example, in a meeting disclosure attack, a typical adversary's objective specifies a pair of users $u$ and $v$, a region $r$, and a time instant $t$, and then it asks whether this pair of users have met at that place and time. We compute the probability of this event as the product

$$\Pr\{\boldsymbol{A}_u^t = r \,|\, \mathbf{o}_{\tilde{u}}, \sigma^*(u) = \tilde{u}\} \cdot \Pr\{\boldsymbol{A}_v^t = r \,|\, \mathbf{o}_{\tilde{v}}, \sigma^*(v) = \tilde{v}\} \tag{3.17}$$

by using the results of the localization attack. A more general attack would specify only a pair of users and ask for the expected number of time instants that they have met in any region. Such questions can be answered by using the results of the localization attack and by computing the expected value of (3.17).

## 3.3 Tracking Attack

In a tracking attack, the adversary's objective is to reconstruct complete or partial actual traces, i.e., he is interested in the *sequences* of events, rather than isolated events (e.g., in a localization attack). To perform this attack, the adversary needs to have some knowledge on each user's transition probability between locations, otherwise the attack is not possible. Hence, we present an algorithm for the tracking attack given that the adversary knowledge is $(\hat{\boldsymbol{\pi}}, \hat{\mathbf{p}})$.

We define the objective of this attack to be finding the most likely trace for each user. Given $\boldsymbol{\sigma}^*$ as the result of the de-anonymization attack (see Section 3.1), the tracking can be formalized as finding

$$\hat{\mathbf{a}}_u^* = \arg\max_{\hat{\mathbf{a}}_u \in \mathcal{A}_u} \Pr\{\hat{\mathbf{a}}_u \,|\, \mathbf{o}_{\tilde{u}}, \sigma^*(u) = \tilde{u}, \hat{\mathbf{p}}\}. \tag{3.18}$$

We use the Viterbi algorithm [Rab89], which is a dynamic programming algorithm, in order to perform this attack. Let $\delta_r^u(t)$ be

$$\delta_r^u(t) = \max_{\hat{\mathbf{a}}_u^{1:t-1}} \Pr\{\hat{\mathbf{a}}_u^{1:t-1}, \boldsymbol{A}_u^t = r, \mathbf{o}_{\tilde{u}}^{1:t-1} \,|\, \sigma^*(u) = \tilde{u}, \hat{\mathbf{p}}\}, \tag{3.19}$$

which is the joint probability of the most likely trace of user $u$ in the time windows $[1, t]$ that the user's location at time $t$ is $r$, and her trace observed up to time $t$, which is $\mathbf{o}_{\tilde{u}}^{1:t-1}$. Maximizing this quantity at time $T$ is equivalent to maximizing (3.18). We recursively compute its values, for $2 \le t \le T$, as following:

$$\begin{aligned}\delta_r^u(t) = &\max_{r' \in \mathcal{R}} \big(\delta_{r'}^u(t-1) \cdot \underbrace{\Pr\{\boldsymbol{A}_u^t = r \,|\, \boldsymbol{A}_u^{t-1} = r', \hat{\mathbf{p}}\}}_{\text{User Mobility Profile}}\big) \\ &\cdot \sum_{x \in \{0,1\}} \underbrace{\Pr\{o_{\tilde{u}}^t \,|\, \boldsymbol{X}_u^t = x, \boldsymbol{A}_u^t = r, \sigma^*(u) = \tilde{u}, \hat{\mathbf{p}}\}}_{\text{LPPM - Obfuscation mechanism}} \cdot \underbrace{\Pr\{\boldsymbol{X}_u^t = x \,|\, \boldsymbol{A}_u^t = r, \hat{\mathbf{p}}\}}_{\text{LBS Application}}.\end{aligned} \tag{3.20}$$

The initialization at time $t = 1$ is

$$\begin{aligned}\delta_r^u(1) = &\underbrace{\Pr\{\boldsymbol{A}_u^1 = r \,|\, \hat{\mathbf{p}}\}}_{\text{User Mobility Profile}} \\ &\cdot \sum_{x \in \{0,1\}} \underbrace{\Pr\{o_{\tilde{u}}^1 \,|\, \boldsymbol{X}_u^1 = x, \boldsymbol{A}_u^1 = r, \sigma^*(u) = \tilde{u}, \hat{\mathbf{p}}\}}_{\text{LPPM - Obfuscation mechanism}} \cdot \underbrace{\Pr\{\boldsymbol{X}_u^1 = x \,|\, \boldsymbol{A}_u^1 = r, \hat{\mathbf{p}}\}}_{\text{LBS Application}}.\end{aligned} \tag{3.21}$$

From the values $\delta_r^u(T)$, we compute the joint probability of the most likely trace and the observations by computing

$$\max_{r \in \mathcal{R}} \delta_r^u(T). \tag{3.22}$$

Of course, we are interested in the most likely trace itself, not only in its probability. The most likely trace is computed by keeping track, at each time $2 \leq t \leq T$, of the location that maximizes (3.20) and, for $t = T$, the one that maximizes (3.22). Then, we can backtrack from time $T$ back to time 1 and reconstruct the trace.

Notice that finding the most likely trace is exactly equivalent to finding the shortest path in an edge-weighted directed graph. The graph's $MT$ vertices are labeled with elements of the set $\mathcal{R} \times \mathcal{T}$, i.e., for each time $t$ there are $M$ vertices corresponding to each of the $M$ regions. There are edges only from vertices labeled with time $t$ to vertices labeled $t+1, 1 \leq t < T$. The weight of an edge $(t-1, r') \to (t, r)$ is equal to

$$\begin{aligned} -\log \big( & \underbrace{\mathbb{P}\mathrm{r}\{\boldsymbol{A}_u^t = r \,|\, \boldsymbol{A}_u^{t-1} = r', \hat{\mathbf{p}}\}}_{\text{User Mobility Profile}} \\ & \cdot \sum_{x \in \{0,1\}} \underbrace{\mathbb{P}\mathrm{r}\{o_{\tilde{u}}^t \,|\, \boldsymbol{X}_u^t = x, \boldsymbol{A}_u^t = r, \sigma^*(u) = \tilde{u}, \hat{\mathbf{p}}\}}_{\text{LPPM - Obfuscation mechanism}} \cdot \underbrace{\mathbb{P}\mathrm{r}\{\boldsymbol{X}_u^t = x \,|\, \boldsymbol{A}_u^t = r, \hat{\mathbf{p}}\}}_{\text{LBS Application}} \big). \end{aligned} \tag{3.23}$$

Indeed, minimizing the sum of negative logarithmic terms is equivalent to maximizing the product of the original probabilities.

## 3.4 A Generic Location Disclosure Attack

We now consider the most general type of location disclosure attack, on which we can compute the *distribution* of traces for each user, rather than simply the most likely trace:

$$\mathbb{P}\mathrm{r}\{\boldsymbol{A} = \hat{\mathbf{a}}, \boldsymbol{X} = \hat{\mathbf{x}}, \boldsymbol{\Sigma} = \hat{\boldsymbol{\sigma}} \,|\, \boldsymbol{O} = \mathbf{o}\} \tag{3.24}$$

In the implementation of this attack, we use the Metropolis-Hastings (MH) algorithm [Has70, Mac03] on the product of the space $\mathcal{A} \times \mathcal{X}$ with the space of all possible permutations $\boldsymbol{\sigma}$. The purpose of the MH algorithm is to draw independent samples (from the space $\mathcal{A} \times \mathcal{X} \times \Sigma$) that are identically distributed according to the desired distribution (3.24). The algorithm makes use of the fact that the desired distribution, briefly written as $\mathbb{P}\mathrm{r}\{\hat{\mathbf{a}}, \hat{\mathbf{x}}, \hat{\boldsymbol{\sigma}} \,|\, \mathbf{o}\}$, is equivalently

$$\mathbb{P}\mathrm{r}\{\hat{\mathbf{a}}, \hat{\mathbf{x}}, \hat{\boldsymbol{\sigma}} \,|\, \mathbf{o}\} = \frac{\mathbb{P}\mathrm{r}\{\hat{\mathbf{a}}, \hat{\mathbf{x}}, \hat{\boldsymbol{\sigma}}, \mathbf{o}\}}{\mathbb{P}\mathrm{r}\{\mathbf{o}\}} = \frac{\mathbb{P}\mathrm{r}\{\mathbf{o} \,|\, \hat{\mathbf{a}}, \hat{\mathbf{x}}, \hat{\boldsymbol{\sigma}}\} \cdot \mathbb{P}\mathrm{r}\{\hat{\boldsymbol{\sigma}} \,|\, \hat{\mathbf{a}}, \hat{\mathbf{x}}\} \cdot \mathbb{P}\mathrm{r}\{\hat{\mathbf{x}} \,|\, \hat{\mathbf{a}}\} \cdot \mathbb{P}\mathrm{r}\{\hat{\mathbf{a}}\}}{\mathbb{P}\mathrm{r}\{\mathbf{o}\}} \tag{3.25}$$

The denominator is a normalizing factor that is hard to compute, however, the algorithm allows us to sample from the distribution $\mathbb{P}\mathrm{r}\{\hat{\mathbf{a}}, \hat{\mathbf{x}}, \hat{\boldsymbol{\sigma}} \,|\, \mathbf{o}\}$ without computing the denominator $\mathbb{P}\mathrm{r}\{\mathbf{o}\}$. Nevertheless, the numerator needs to be easy to compute, which is true in our case: We compute the probability $\mathbb{P}\mathrm{r}\{\mathbf{o} \,|\, \hat{\mathbf{a}}, \hat{\mathbf{x}}, \hat{\boldsymbol{\sigma}}\}$ using the probability mass function of the LPPM obfuscation mechanism; the probability $\mathbb{P}\mathrm{r}\{\hat{\boldsymbol{\sigma}} \,|\, \hat{\mathbf{a}}, \hat{\mathbf{x}}\}$ is constant and equal to $\frac{1}{N!}$, as we use random permutation as the anonymization function; we compute the probability $\mathbb{P}\mathrm{r}\{\hat{\mathbf{x}} \,|\, \hat{\mathbf{a}}\}$ from the LBS application probability mass function; and we compute the probability $\mathbb{P}\mathrm{r}\{\hat{\mathbf{a}}\}$ from the users' mobility profiles.

At a high level, the MH algorithm performs a random walk on the space of possible values for $(\hat{\mathbf{a}}, \hat{\mathbf{x}}, \hat{\boldsymbol{\sigma}})$. The transition probabilities of the random walk are chosen so that its stationary distribution is the distribution from which we want to sample.

First of all, we need to find a *feasible* initial point for the walk (i.e., a point that does not violate the mobility profile of any user and is consistent with the observations from the users; it is not a trivial matter to find such a point). We use the most likely trace of the users, i.e., $\hat{\mathbf{a}}^*$, which is the output of the tracking attack in Section 3.3, as the starting point.

We then need to define a neighborhood for each point. We define two points $(\hat{\mathbf{a}}, \hat{\mathbf{x}}, \hat{\boldsymbol{\sigma}})$ and $(\hat{\mathbf{a}}', \hat{\mathbf{x}}', \hat{\boldsymbol{\sigma}}')$ to be neighbors if and only if exactly one of the four following conditions holds:

- The components $\hat{\boldsymbol{\sigma}}$ and $\hat{\boldsymbol{\sigma}}'$ differ in exactly two places. That is, $N-2$ out of the $N$ traces are assigned to the same users in both $\hat{\boldsymbol{\sigma}}$ and $\hat{\boldsymbol{\sigma}}'$, and the assignment of the remaining two traces to users is switched. The components $\hat{\mathbf{a}}$ and $\hat{\mathbf{a}}'$ are identical, and so are components $\hat{\mathbf{x}}$ and $\hat{\mathbf{x}}'$.

- The components $\hat{\mathbf{a}}$ and $\hat{\mathbf{a}}'$ differ in exactly one place. That is, the location of exactly one user at exactly one time instant is different. All other locations are unchanged. The other components are identical.

- The components $\hat{\mathbf{x}}$ and $\hat{\mathbf{x}}'$ differ in exactly one place. That is, LBS access bit of one user at only one time instant is different. All other bits of $\hat{\mathbf{x}}$ and $\hat{\mathbf{x}}'$ are the same. The other components are identical.

- Points $(\hat{\mathbf{a}}, \hat{\mathbf{x}}, \hat{\boldsymbol{\sigma}})$ and $(\hat{\mathbf{a}}', \hat{\mathbf{x}}', \hat{\boldsymbol{\sigma}}')$ are identical. That is, a point is assumed to be included in its own neighborhood.

We finally define a proposal density function that determines the candidate neighbor to move to, at the next step; this function also influences the convergence speed of the algorithm. For simplicity, we use a uniform proposal density, so the candidate point is selected randomly among all the neighbors of the current point.

To perform the random walk, suppose that the current point is $(\hat{\mathbf{a}}, \hat{\mathbf{x}}, \hat{\boldsymbol{\sigma}})$ and the selected candidate is $(\hat{\mathbf{a}}', \hat{\mathbf{x}}', \hat{\boldsymbol{\sigma}}')$. Then, the candidate point is accepted with the following probability:

$$\min\{1, \frac{\mathbb{P}\mathrm{r}\{\hat{\mathbf{a}}', \hat{\mathbf{x}}', \hat{\boldsymbol{\sigma}}' \,|\, \mathbf{o}\}}{\mathbb{P}\mathrm{r}\{\hat{\mathbf{a}}, \hat{\mathbf{x}}, \hat{\boldsymbol{\sigma}} \,|\, \mathbf{o}\}} = \frac{\mathbb{P}\mathrm{r}\{\mathbf{o} \,|\, \hat{\mathbf{a}}', \hat{\mathbf{x}}', \hat{\boldsymbol{\sigma}}'\} \cdot \mathbb{P}\mathrm{r}\{\hat{\boldsymbol{\sigma}}' \,|\, \hat{\mathbf{a}}', \hat{\mathbf{x}}'\} \cdot \mathbb{P}\mathrm{r}\{\hat{\mathbf{x}}' \,|\, \hat{\mathbf{a}}'\} \cdot \mathbb{P}\mathrm{r}\{\hat{\mathbf{a}}'\}}{\mathbb{P}\mathrm{r}\{\mathbf{o} \,|\, \hat{\mathbf{a}}, \hat{\mathbf{x}}, \hat{\boldsymbol{\sigma}}\} \cdot \mathbb{P}\mathrm{r}\{\hat{\boldsymbol{\sigma}} \,|\, \hat{\mathbf{a}}, \hat{\mathbf{x}}\} \cdot \mathbb{P}\mathrm{r}\{\hat{\mathbf{x}} \,|\, \hat{\mathbf{a}}\} \cdot \mathbb{P}\mathrm{r}\{\hat{\mathbf{a}}\}}\}. \tag{3.26}$$

In the explanation of (3.25), we described that these probabilities can easily be computed given the knowledge of LPPM, LBS, and user mobility profile. Moreover, as the two neighbors differ only in a small number of elements, the acceptance probability can become even simpler by cancelling out the identical probabilities in the numerator and denominator of (3.26).

If $(\hat{\mathbf{a}}', \hat{\mathbf{x}}', \hat{\boldsymbol{\sigma}}')$ is rejected, we log the current step, and we repeat the procedure of selecting and probabilistically moving to a neighbor. If it is accepted, it is logged as a step in the random walk. However, the logged sample is not an independent sample, as it is correlated with $(\hat{\mathbf{a}}, \hat{\mathbf{x}}, \hat{\boldsymbol{\sigma}})$. Only after making enough steps to overcome the inherent correlation among successive steps is a step stored as an independent sample. After storing enough independent samples, the algorithm stops.

How many independent samples are enough? The attacker collects as many samples as needed to satisfy his accuracy requirements. The confidence interval for the chosen confidence level must be shorter than the desired length. Suppose the attacker needs to collect $n$ independent samples.

How many steps of the random walk must be taken between each pair of successive samples to ensure the independence of these $n$ samples? There are standard statistical tests of independence; our choice is the *Turning Point* test [LB10]. The basic idea of this test is that, among three successive independent and identically distributed samples, all $3! = 6$ possible orderings are equiprobable. Given three numerical values $z_{i-1}, z_i, z_{i+1}$, a *turning point* exists at $i$ if and only if $z_i$ is either larger than both $z_{i-1}, z_{i+1}$ or smaller than both $z_{i-1}, z_{i+1}$. If the three numerical values are independent and identically distributed, then the probability of a turning point is $\frac{2}{3}$. Then, given a large enough number of values, $n$ in our case, the number of turning points is approximately Gaussian with mean $\frac{2n-4}{3}$ and variance $\frac{16n-29}{90}$.

So, we test if the number of turning points in our sequence of $n$ MH samples can be claimed to be Gaussian with this mean and variance. If so, we stop. Otherwise, we make more steps in the random walk and skip more of the logged intermediate steps before storing each sample.

It should be emphasized that, after collecting enough number of samples, this generic attack can answer all kinds of U-R-T questions. The attacker can specify a very wide range of objectives as functions of a sample of the MH algorithm. Then, the attacker computes this function on each independent sample, and the sample average of the computed values is the estimate of the attacker's objective. In this case, the accuracy and certainty metrics would be computed on the values that the function returns, rather than directly on the MH samples.

## 3.5 Evaluation

We implement the user profiling algorithms (presented in Chapter 2) and the inference attacks (presented in this chapter) in a tool called *Location Privacy and Mobility Meter* (LPM). The tool enables us to evaluate various location-privacy preserving mechanisms with respect to various location inference attacks. See [LPM12] for more details about our software tool. In this section,

- we show a few examples of using $LPM$ to quantify the effectiveness of LPPMs to protect users' anonymity and location privacy.
- we evaluate the appropriateness of two popular metrics, namely, *k-anonymity* and *entropy*, for quantifying location privacy.

### 3.5.1 LPM Simulation Setting

**Location Traces** The location traces that we use are interpolated traces of $N = 20$ randomly chosen mobile users (vehicles) from the epfl/mobility dataset at CRAWDAD [PSDG]. Each trace contains the location of a user every 5min for a full day, $T = 288$. The location area (the San Francisco bay area) is divided into $M = 40$ regions forming a $5 \times 8$ grid.

**LBS** We assume that at any time instant, a user accesses the LBS with some probability $\theta$, independently from other users. More formally, $\Pr\{\boldsymbol{X}_u^t = 1 \,|\, \boldsymbol{A}_u^t = r\} = \theta$. This implicitly gives more weight to the locations that a user visits more frequently.[3]

[3] We could consider alternative LBS access patterns as well. However, our goal here is not to evaluate all different types of LBSs. That is why we only consider a simple model for LBS.

**User Profiling** To consider the worst case scenario, we use the users' actual traces as the adversary's prior information on the users' mobility patterns. Then, we run the *user profiling* algorithm, presented in Chapter 2, to construct both the transition probability matrix $\hat{\mathbf{p}}$, and the location probability distribution $\hat{\boldsymbol{\pi}}$, for each user.

**Inference Attacks** We run the de-anonymization and localization attacks on the users' observed traces, and obtain results for the following attack scenarios:

- *De-Anonymization Attack*: For all users being observed until time $t$, what is the user identity associated to each observed trace (pseudonym)?
- *Localization Attack*: For a given user $u$ and time $t$, what is the location of $u$ at $t$? (As the user's location is a random variable, the answer is a probability distribution over the locations).
- *Meeting (Location Proximity) Disclosure Attack*: For a given pair of users $u$ and $v$, what is the expected number of meetings between $u$ and $v$? Put differently, at how many time instants in $\mathcal{T}$ the two users are in the same region (in the proximity of each other).
- *Aggregated Presence Disclosure Attack*: For a given region $r$ and time $t$, what is the expected number of users present in $r$ at $t$?

**LPPM** The protection mechanism, that we evaluate, anonymizes the traces by using a random permutation function (i.e., each user is assigned a unique and permanent pseudonym randomly chosen from 1 to $N$). For our considered location obfuscation methods, we have to define two modes of behavior, according to whether the user accesses the LBS (i.e., exposes her location). When the user accesses the LBS, the LPPM obfuscates her location by removing some low-order bits/digits of the location-stamp of the event. We refer to the number of removed bits as the *obfuscation level* $\rho$ of the LPPM. So, for a given $\rho$, the user's location is obfuscated among $2^\rho$ location. In the case when the user does not access the LBS, the LPPM chooses, with some probability $\phi$, to create a fake location and then obfuscates it (as it does for the actual locations). We consider two ways in which the LPPM can create a fake location: The first way is to create a fake location uniformly at random among all locations $r \in \mathcal{R}$, and the second way is to create it according to the aggregate user geographical distribution $\bar{\pi} = \frac{1}{N}\sum_u \hat{\boldsymbol{\pi}}_u$ (i.e., the average location probability distribution). We refer to an LPPM using its parameters $\phi$ and $\rho$, and its type (u: uniform selection, g: selection according to the average mobility profile). For example $\mathsf{LPPM}(\mathrm{A}, \mathrm{O2}, \mathrm{F0.3u})$ anonymizes the traces, injects a fake location (uniformly selected at random) with probability 0.3 if there is no LBS access, and obfuscates the (both fake and actual) locations by dropping their 2 low-order bits, whereas $\mathsf{LPPM}(\mathrm{A})$ only anonymizes the traces.

**Privacy Metrics** The metric that we use to evaluate the LPPMs is the expected error (incorrectness) of the adversary, as described in Section 1.5.1. We evaluate the effect of the LPPM parameters that we listed above (obfuscation level, probability of fake location injection, different ways of creating fake locations) on the users' privacy. We are also interested in the effect of the pseudonym lifetime on the anonymity of users. In our model, we consider that all users keep their pseudonyms from time 1 to $T$. By attacking at time $T$, we can compare the anonymity achieved by users for various values of $T$.

The *anonymity* metric is the fraction of incorrectly de-anonymized observed traces, hence its values range from 0 to 1. In the case of *localization* attack, we compute the location privacy of each user $u$ at each time instant $t$ as the adversary's probability of error (i.e., we use Hamming distortion function in computing the expected estimation error) that is $1 - \hat{\mathbb{P}}\mathrm{r}\{\boldsymbol{A}_u^t = a_u^t \,|\, \mathbf{o}\}$, hence its values range from 0 to 1. To quantify users' privacy against *meeting (location proximity) disclosure* attack, we compute the error of attacker in estimating the number of meetings between two users $u$ and $v$ over all time instants as

$$\Big| \sum_t 1_{a_u^t = a_v^t} - \sum_t \sum_r \hat{\mathbb{P}}\mathrm{r}\{\boldsymbol{A}_u^t = r \,|\, \mathbf{o}\} \cdot \hat{\mathbb{P}}\mathrm{r}\{\boldsymbol{A}_v^t = r \,|\, \mathbf{o}\} \Big| \tag{3.27}$$

whose values range from 0 to $T$. In the case of *aggregated presence disclosure* attack, the attacker estimates the expected number of users in each region $r$ at each time $t$ as the summation $\sum_u \hat{\mathbb{P}}\mathrm{r}\{\boldsymbol{A}_u^t = r \,|\, \mathbf{o}\}$, hence we compute his error (i.e., the users' privacy) as

$$\Big| \sum_u 1_{a_u^t = r} - \sum_u \hat{\mathbb{P}}\mathrm{r}\{\boldsymbol{A}_u^t = r \,|\, \mathbf{o}\} \Big| \tag{3.28}$$

whose values range from 0 to $N$.

### 3.5.2 Simulation Results

We run the LPM for various combinations of the following parameters:

- LBS access probabilities $\{0.1, 0.2, ..., 1.0\}$
- location obfuscation levels $\{0, 2, 4\}$
- fake-location injection rates $\{0, 0.3, 0.6\}$ with uniform selection u and general mobility selection g
- observation time period $\{31, 71, 141, 281\}$

We then perform the de-anonymization and localization attacks (for the adversary knowledge being $\hat{\boldsymbol{\pi}}$ or $(\hat{\boldsymbol{\pi}}, \hat{\mathbf{p}})$). The results are averaged over 20 simulation runs. Hereafter, we present some of the results that we obtained regarding the anonymity and location-privacy of users.

#### Anonymity

In Figure 3.1, we plot user anonymity as a function of pseudonym lifetime (how long the users' accesses to the LBS are observed by the adversary). The anonymity is quantified as the percentage of users that are incorrectly de-anonymized by the attacker. Each of the four sub-figures corresponds to each of the four combinations of adversary knowledge and LPPMs. Each line in a sub-figure corresponds to different combinations of obfuscation levels and probabilities of injecting a fake location.

We observe that the anonymity decreases as the length of the observation period increases. The same trend is seen in all four sub-figures, for all combination parameters. By comparing the results that are obtained from different LPPMs, we observe the following interesting phenomenon, regarding the effect of stronger LPPM parameters, in particular when both the obfuscation level and the fake injection probability are non-zero: By jointly increasing the

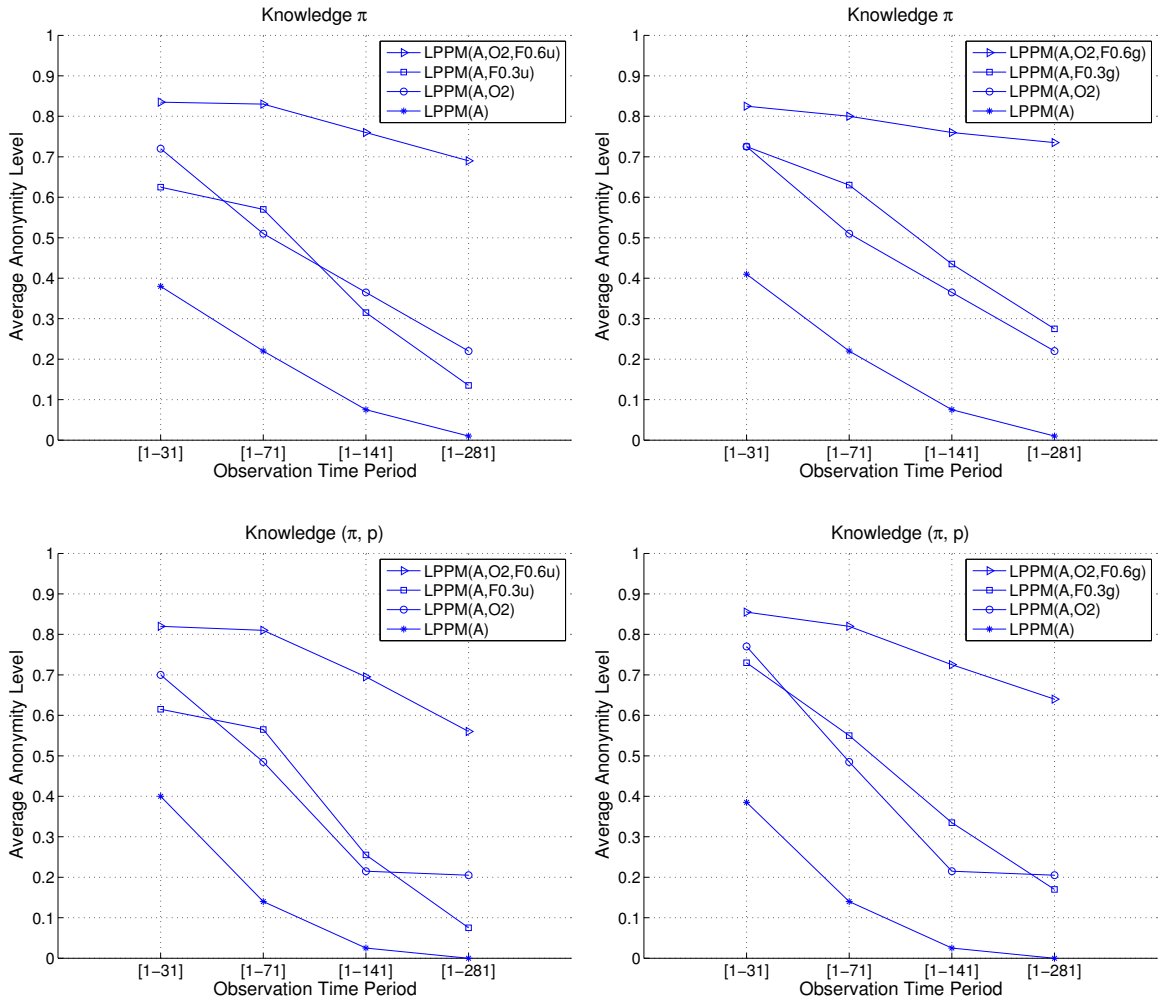



Figure 3.1: User **anonymity** versus pseudonym lifetime. Users access the LBS sporadically with probability 0.1. The anonymity is quantified as the percentage of users that are incorrectly de-anonymized by the attacker. In the top two sub-figures, we consider the adversary with his background knowledge being $\hat{\boldsymbol{\pi}}$, whereas in the bottom two, we consider a more knowledgeable adversary with his background knowledge being $(\hat{\boldsymbol{\pi}}, \hat{\mathbf{p}})$. The fake-location injection algorithm used in the left column is the uniform selection (u), whereas the right column considers the general mobility fake-location selection (g) according to the average location distribution $\bar{\pi}$. Each line in a sub-figure corresponds to different combinations of obfuscation levels $\{0, 2\}$ and fake-location injection rates $\{0, 0.3, 0.6\}$. All the LPPMs used here anonymize the traces.

protection level of the two mechanisms, the absolute value of anonymity increases. Additionally, the level of anonymity drops with a slower rate as the pseudonym lifetime increases. This shows the relation between the effects of obfuscation and anonymization techniques. The LPPM designer can appropriately choose the parameters to achieve a desired level of anonymity; or alternatively, the pseudonym should be changed when the desired level of anonymity is no longer achieved.

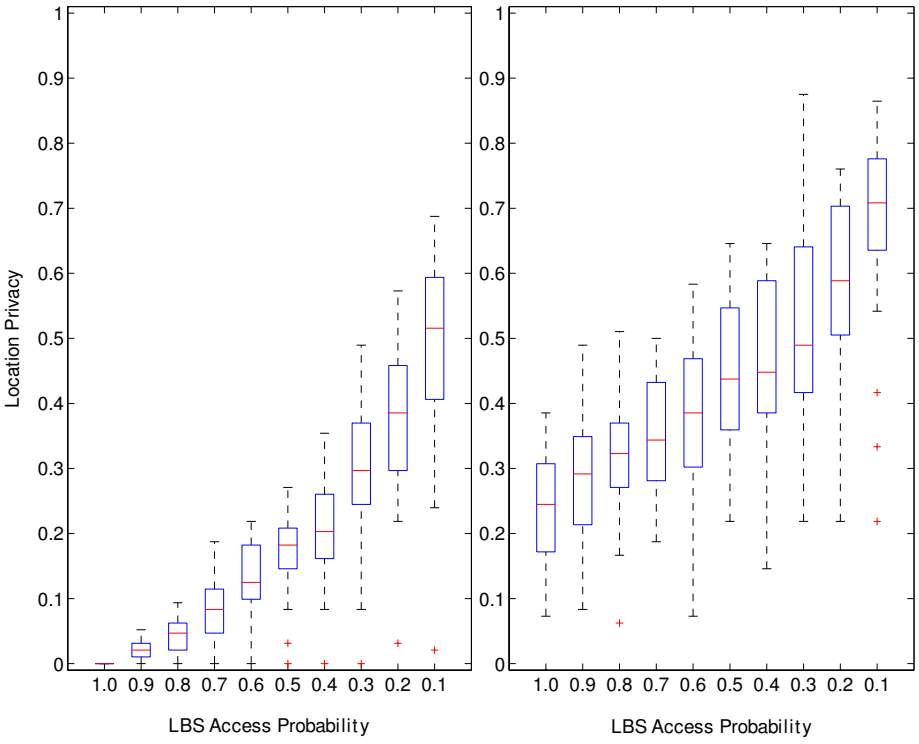



Figure 3.2: System-level location privacy against **localization attack**, for all users $u$ and times $t$. Left-hand and right-hand side plots, respectively, show the attack results against LPPM(A), that only anonymizes the traces, and LPPM(A, O4), that in addition to anonymization it also obfuscates the users' locations among $2^4$ locations when they access the LBS. The $x$-axis shows the probability that a user accesses the LBS at any time instant. By moving from the left side to the right side in each plots, the amount of information shared with the attacker (through his observed events) decreases, as the user accesses the location-based service less frequently. The considered attacker's background knowledge here is $(\hat{\boldsymbol{\pi}}, \hat{\mathbf{p}})$. In the boxplots, the bottom and top of each box show the $25^{th}$ and $75^{th}$ percentiles, respectively, and the central mark shows the median. The ends of the whiskers represent the most extreme data points not considered as outliers, and the outliers are plotted individually.

### Location Privacy

Figure 3.2 illustrates the results that we have obtained about the effectiveness of the anonymization and precision-reduction location obfuscation against the localization attacks. The left-hand plot shows the results for the LPPM with anonymization, and the right-hand plot shows the results for the LPPM with location obfuscation level 4. The probability at which users access the LBS varies from 1.0 (for a fully continuous LBS) to 0.1 (for a sporadic LBS). As we expect, users' location privacy increases as their access frequency to the LBS decreases. Obfuscating their locations by hiding them among almost half of the locations also leads to a non-negligible increase in their location privacy. Looking at the two ends of these plots, we note the following interesting points. In Figure 3.2(a), users have zero location privacy when their privacy is protected only through anonymization and when they access the LBS

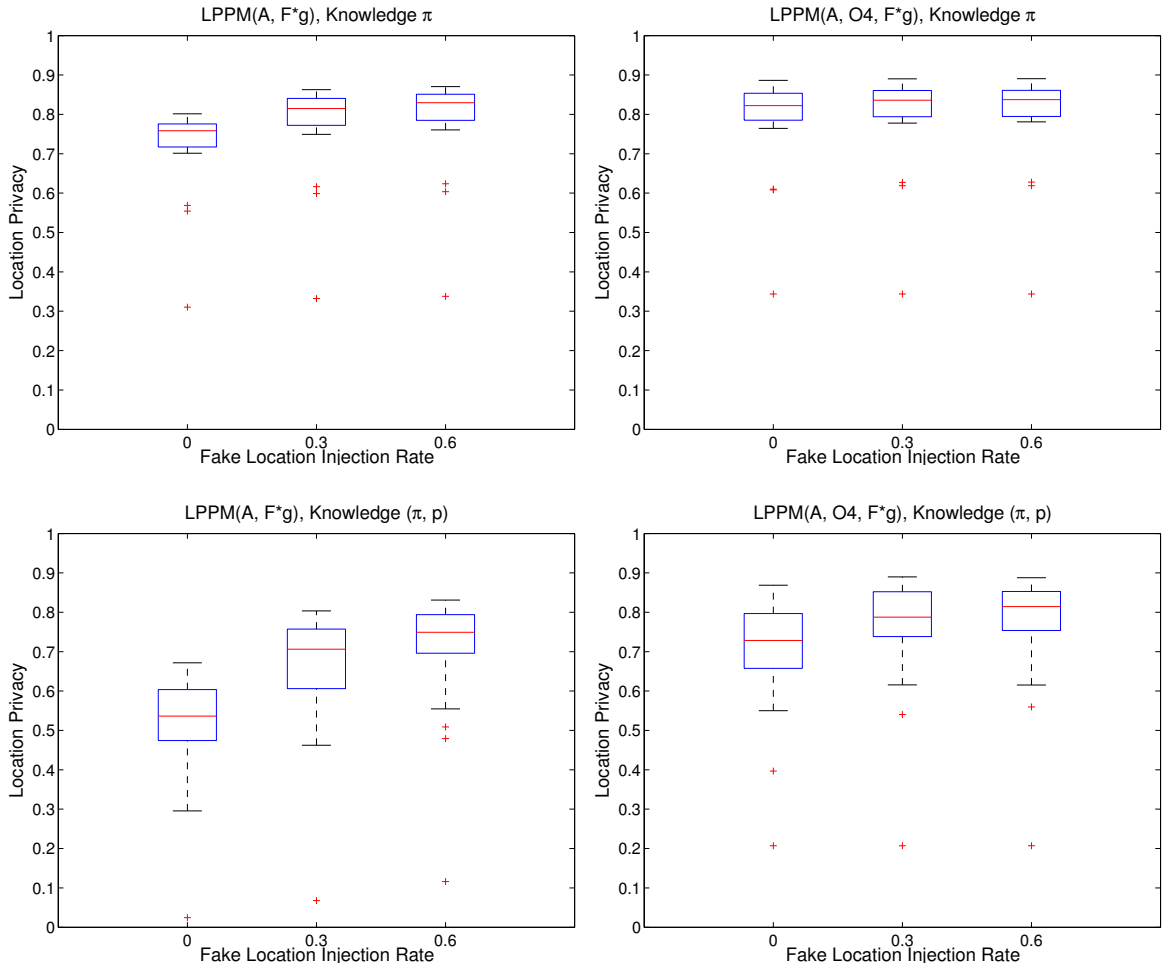



Figure 3.3: Users' location privacy in *sporadic* LBSs (accessing the LBS with probability 0.1 at any time instant), using various LPPMs, with respect to **localization attack** performed by the attacker with two different classes of background knowledge (first row: $\hat{\boldsymbol{\pi}}$, and second row: $(\hat{\boldsymbol{\pi}}, \hat{\mathbf{p}})$). The tested LPPMs anonymize the traces, and inject fake locations when a user does not access the LBS. The sub-figures corresponds to LPPM with obfuscation level 4 (for the two right side plots), and 0 (for the two left side plots). The x-axis shows the fake-location injection rate $\phi$ of the LPPMs. Each box-and-whisker diagram (boxplot) shows system level location-privacy, where the bottom and top of each box show the $25^{th}$ and $75^{th}$ percentiles, and the central mark shows the median. The ends of the whiskers represent the most extreme data points not considered as outliers, and the outliers are plotted individually.

continuously (when $x$-axis value is 1.0). In Figure 3.2(b), for the case of the most shown sporadic LBS, when users' LBS access probability is 0.1, despite the trace anonymization and the high degree of location obfuscation, the users' location privacy is still 30 percent below its maximum value 1. This is, due to the predictability of users' locations given their mobility profiles, that enables the adversary to still estimate their locations over time, although there is not much information in their observed traces.

Now, let us focus on a sporadic LBS and study the effects of more location-privacy preserv-

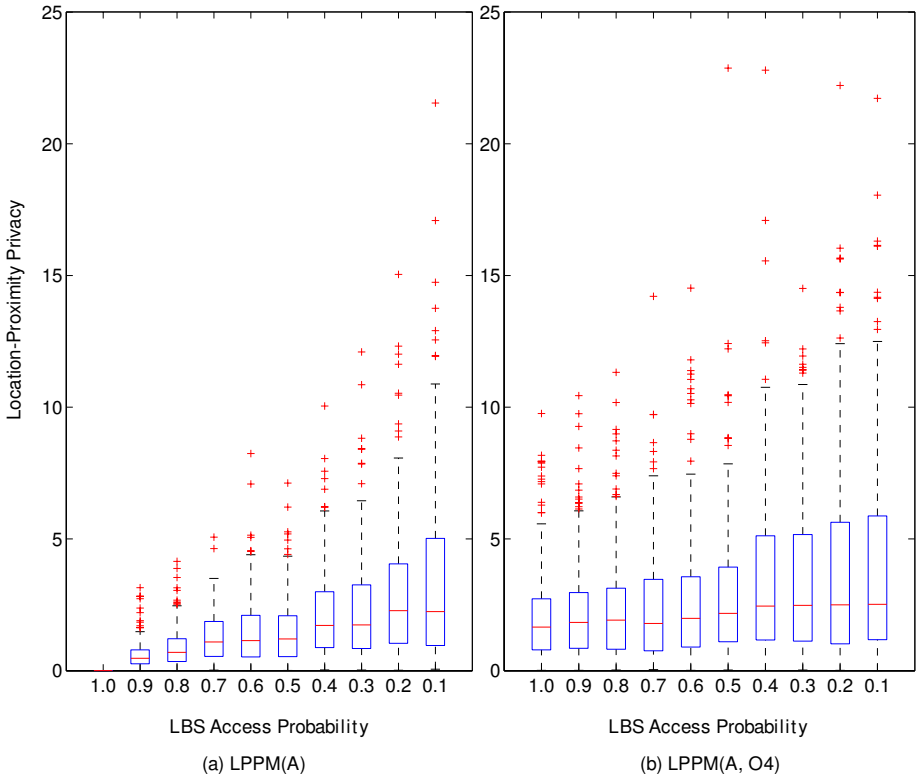



Figure 3.4: System-level location privacy against **meeting (location proximity) disclosure attack**, for all pairs of users $u, v$. Left-hand and right-hand side plots, respectively, show the attack results against LPPM(A), that only anonymizes the traces, and LPPM(A, O4), that in addition to anonymization it also obfuscates the users' locations among $2^4$ locations when they access the LBS. The $x$-axis shows the probability that a user accesses the LBS at any time instant. By moving from the left side to the right side in each plot, the amount of information shared with the attacker (through his observed events) decreases, as the user accesses the location-based service less frequently. The considered attacker's background knowledge here is $(\hat{\boldsymbol{\pi}}, \hat{\mathbf{p}})$. In the boxplots, the bottom and top of each box show the $25^{th}$ and $75^{th}$ percentiles, respectively, and the central mark shows the median. The ends of the whiskers represent the most extreme data points not considered as outliers, and the outliers are plotted individually.

ing mechanisms on the expected error of the adversary in localization attacks. In Figure 3.3, we show the location privacy of users who sporadically access the LBS (with access probability 0.1), and use the LPPM that additionally adds fake locations to the users' observed traces according to the *aggregate user geographical distribution* $\bar{\pi}$, when users do not access the LBS. As it is expected, the users' location privacy increases when the level of location-obfuscation or fake-location injection increases. However, the main finding of our result is that, in sporadic applications, the fake-location injection can considerably help the obfuscation method in preserving users' location-privacy, when the injection rate is high. Moreover, adding fake location has a high impact on misleading the more knowledgeable adversary (with knowledge $(\hat{\boldsymbol{\pi}}, \hat{\mathbf{p}})$, as it reduces his success down to that of the adversary with knowledge $\hat{\boldsymbol{\pi}}$ (compare the location-privacy improvement obtained by injecting fake-locations with rate 0.3 in the

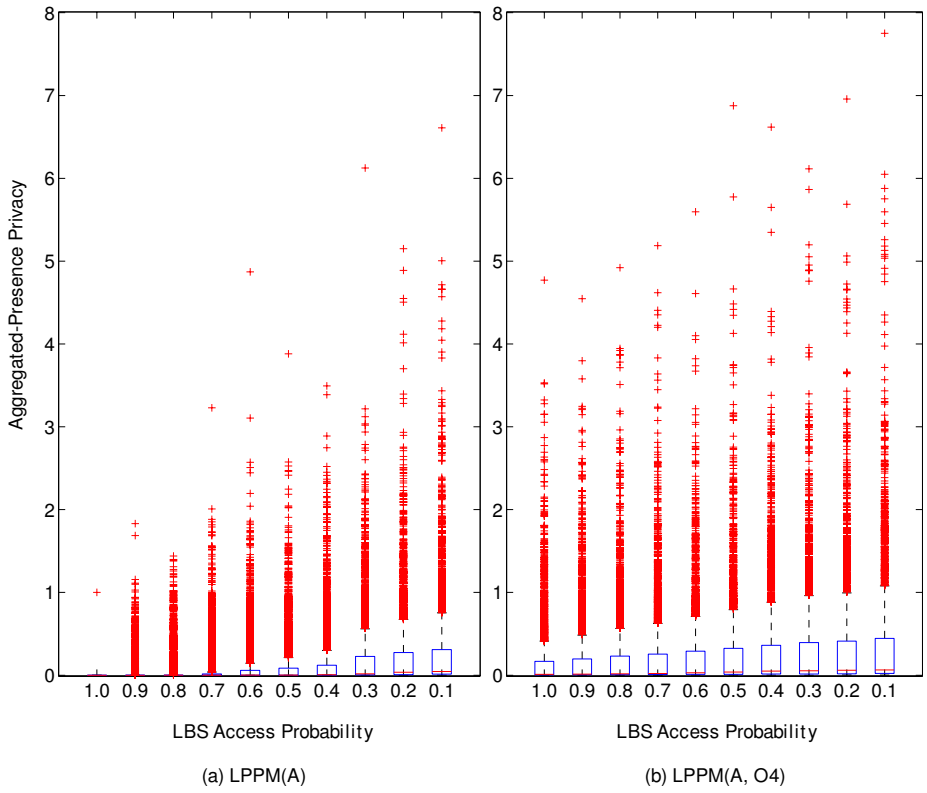
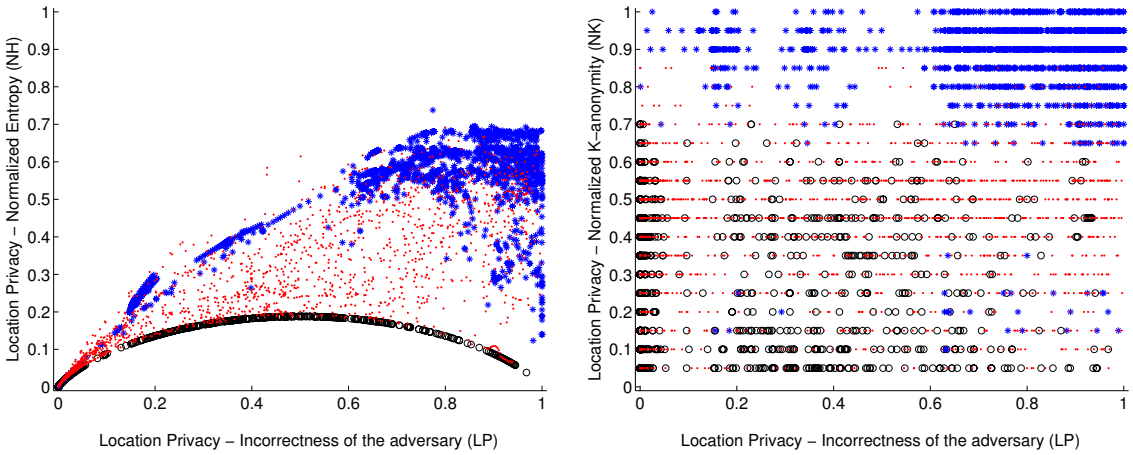



Figure 3.5: System-level location privacy against **aggregated presence disclosure attack**, for all regions $r$ and times $t$. Left-hand and right-hand side plots, respectively, show the attack results against LPPM(A), that only anonymizes the traces, and LPPM(A, O4), that in addition to anonymization it also obfuscates the users' locations among $2^4$ locations when they access the LBS. The $x$-axis shows the probability that a user accesses the LBS at any time instant. By moving from the left side to the right side in each plots, the amount of information shared with the attacker (through his observed events) decreases, as the user accesses the location-based service less frequently. The considered attacker's background knowledge here is $(\hat{\boldsymbol{\pi}}, \hat{\mathbf{p}})$. In the boxplots, the bottom and top of each box show the $25^{th}$ and $75^{th}$ percentiles, respectively, and the central mark shows the median. The ends of the whiskers represent the most extreme data points not considered as outliers, and the outliers are plotted individually.

bottom sub-figures).

Figure 3.4 and 3.5 illustrate the users' privacy with respect to meeting (location proximity) disclosure attack and aggregated presence disclosure attack, respectively. As in Figure 3.2, we expect to see improvement in location privacy, as we increase the level of obfuscation. We also expect to observe convergence of location privacy to its near maximum value, when we set the LBS access probability equal to 0.1 (i.e., 90% of the users' locations are hidden from the adversary). Unsurprisingly, we observe these two things in the plots: Reading a plot from left to right, we see the effect of decreasing the access probability (1.0 to 0.1) for constant precision-reducing levels. Specifically, the privacy always increases, although the effect is much more pronounced in Figure 3.2. By comparing corresponding boxes of two adjacent plots, i.e., same LBS access probability, we see the added value of the precision-reducing mechanism (on




(a) Entropy vs. Incorrectness (Distortion) (b) K-anonymity vs. Incorrectness (Distortion)

Figure 3.6: Comparison of location-privacy metrics. The $x$-axis shows the users' location-privacy based on the incorrectness (distortion) metric (1.7), that is the adversary's expected estimation error. The $y$-axis shows (a) the normalized entropy of the adversary's estimation, (b) the normalized k-anonymity of users. Each point in the plot represents the location privacy of some user at some time for two metrics (incorrectness vs entropy in (a), incorrectness vs k-anonymity in (b)). "$*$"s are the location privacy values achieved from $\mathsf{LPPM}(\mathsf{A}, \mathsf{O5})$ as a strong mechanism protecting users' sporadic access to the $\mathsf{LBS}$ with access probability 0.1, "$\cdot$"s are the values for $\mathsf{LPPM}(\mathsf{A}, \mathsf{O3})$ as a medium mechanism protecting users' access to the $\mathsf{LBS}$ with access probability 0.5, and "$\circ$"s are the values for $\mathsf{LPPM}(\mathsf{A}, \mathsf{O1})$ as a weak mechanism protecting users' continuous access to the $\mathsf{LBS}$ with access probability 1.0. The considered attacker's background knowledge in all cases is $(\hat{\boldsymbol{\pi}}, \hat{\mathbf{p}})$. The two metrics would be fully correlated only if all points were on the diagonal $(0, 0)$ to $(1, 1)$.

the left, there is no location obfuscation; on the right, obfuscation level is 4). Again, the clearest improvement happens in Figure 3.2. An interesting conclusion is that the effect of the LPPM is most positive against Figure 3.2, which is, in a sense, the most intrusive attack of the three: it targets the exact location of a single user at a single time. The other two attacks, especially Figure 3.5, are more related to the statistics of user mobility, so there could even be legitimate reasons that one would want to collect that information. For instance, a researcher who studies the geographical distribution of users would be interested in the number of users in a region. We can conclude that the tested $\mathsf{LPPM}$s protect users' location-privacy against malicious adversaries, but they still provide information for less harmful activities.

### Existing Metrics for Location Privacy

Now, we assess the appropriateness of two existing metrics, namely *k-anonymity* and *entropy*, for quantifying location privacy. Note that any other heuristic metric can be evaluated in the same way. We focus on the localization attack, and we assess these metrics by testing to what extent they are correlated to the success of the adversary in correctly localizing users over time (i.e., the incorrectness/distortion/expected-estimation-error metric). We choose three LPPMs: $\mathsf{LPPM}(\mathsf{A}, \mathsf{O1})$ on $\mathsf{LBS}(1.0)$ as a weak mechanism for a continuous LBS, $\mathsf{LPPM}(\mathsf{A}, \mathsf{O3})$ on $\mathsf{LBS}(0.5)$ as a medium-strength mechanism, and $\mathsf{LPPM}(\mathsf{A}, \mathsf{O5})$ on $\mathsf{LBS}(0.1)$ as a strong

mechanism on a sporadic LBS.

In Section 1.5.1, we use entropy to measure the uncertainty of the adversary. Here, we assess the normalized entropy of the probability distribution of the location of user $u$ at time $t$, as a metric for her location privacy. The normalized entropy is computed as follows:

$$NH(u,t) = -\frac{1}{\log(M)} \sum_{r \in \mathcal{R}} \hat{\mathbb{P}}\mathrm{r}\{\boldsymbol{A}_u^t = r \,|\, \mathbf{o}\} \log(\hat{\mathbb{P}}\mathrm{r}\{\boldsymbol{A}_u^t = r \,|\, \mathbf{o}\}) \tag{3.29}$$

where $\log(M)$ is the maximum entropy over $M$ regions.

According to the k-anonymity metric, the location-privacy of a user $u$ at a given time $t$ is equal to the number of users who satisfy all of the following conditions: (i) they are anonymous, (ii) they obfuscate their location by merging regions (which includes their actual location), (iii) their obfuscated location (i.e., the set of merged regions) is a superset of the obfuscated location of $u$ at $t$. We divide this number of users by $N$, the total number of users, to have the normalized k-anonymity:

$$NK(u,t) = \frac{1}{N} \sum_{v \in \mathcal{U}} 1_{a_v^t \in o_u^t \wedge o_u^t \subseteq o_v^t} \tag{3.30}$$

Figure 3.6 illustrates the relation between the incorrectness (distortion) of the adversary $LP(u,t) = 1 - \hat{\mathbb{P}}\mathrm{r}\{\boldsymbol{A}_u^t = a_u^t \,|\, \mathbf{o}\}$ and the two above-mentioned metrics: normalized entropy $NH(u,t)$, and normalized k-anonymity $NK(u,t)$. We see that the entropy is more correlated to the adversary's incorrectness than k-anonymity is. However, both entropy and k-anonymity misestimate the true location privacy of users.

**Entropy Metric** Let us focus on Figure 3.6(a). All but few of the points fall into the "$NH < LP$" triangle, which means that, in this setting, the entropy metric underestimates location privacy. For example, consider the "$*$"s on the $NH = 0.6$ horizontal line, all of whose entropy is 0.6. The incorrectness metric ($LP$) of these points ranges from 0.6 to 1. Or, consider the vertical line $LP = 1$, where there are "$*$"s corresponding to values of $NH$ ranging from 0.2 to 0.7. In both cases, the estimation of location privacy by $NH$ is up to 5 times less than the true location privacy of users, which makes it an inappropriate and loose lower bound for location privacy. We observe the same phenomenon in the results of the two other LPPMs (represented by "$\cdot$"s and "$\circ$"s).

The reason why the entropy metric is not an appropriate metric to quantify location privacy is that it ignores the *true* location of the user. This metric is rather a better metric for the adversary to know how concentrated his *estimation* about the location of the user is. Intuitively, the weaker an LPPM is the more concentrated the adversary's estimation around the user's true location will be. But, for stronger LPPMs, the adversary's estimation will be more concentrated around locations with high prior probability and not the true location. So, the entropy metric is not a good indicator of the user's current privacy at each location.

**k-anonymity Metric** The results are even worse for k-anonymity in Figure 3.6(b), as there is less correlation between $NK$ and $LP$. In fact, k-anonymity in some cases underestimates location privacy (consider the area where $NK < 0.5$ and $LP > 0.5$) and in some other cases ($NK > 0.5$ and $LP < 0.5$) overestimates it. Hence, this is not an appropriate estimator for location privacy either.

K-anonymity metric fails to reflect the correct location privacy of users for multiple reasons. First of all, the level of obfuscation on each location (that contributes to the location privacy of a user) is totally independent of $k$. It rather depends on the relative location of $k-1$ other users who are currently near the user. Secondly, the metric completely ignores the fact that the adversary can invert the obfuscated locations to their true values, given his knowledge on the user's location distribution. Thus, the error of the adversary in his inference attack is not correlated with $k$ determined by k-anonymity metric.

## 3.6 Summary

In this chapter, we have proposed a Bayesian inference scheme to reconstruct users' location information, given what an adversary observes from them, and what he knows a priori about their mobility, their LBS access pattern, and the LPPM that anonymizes and obfuscates their location traces. We formalize the de-anonymization attack as a maximum weight assignment problem and use Bayesian inference applied on hidden Markov models to compute the likelihood of assigning a user to each pseudonym and to consequently solve the problem. Furthermore, given the computed most likely assignment of users to pseudonym (observed traces), we attack the LPPM output to reconstruct the users' locations over time using Bayesian forward-backward and dynamic-programming Viterbi algorithms. Hence, we not only compute the posteriori location probability distribution for each user at each time instant, but also we construct the most-likely (complete) trace for each user. These enable us to quantify users' location privacy, by comparing the attack results with the actual traces of the users.

We have proposed using the Metropolis-Hastings algorithm, which is a Markov Chain Monte Carlo (MCMC) method, in order to compute the a posteriori distribution over the set of possible location traces. This algorithm is based on the creation a Markov chain on the state space of users' location traces, and sampling from this space by running a random walk on that Markov chain. After generating enough samples, we can compute any function, in particular the location privacy metric, on the set of samples as an estimation of the accurate posterior value of that function. We have assumed users' pseudonyms to be persistent over time, i.e., users do not change their pseudonyms during the observation time period. This assumption can be relaxed, especially by extending the Monte Carlo solution scheme, to also evaluate the effectiveness of pseudonym change LPPMs.

In the end, we have shown the results obtained from running the proposed inference attacks, implemented in a software tool called Location Privacy (and Mobility) Meter, on some location traces, assuming some example LBS and LPPMs. We have also shown that existing metrics entropy and k-anonymity are not appropriate metrics for quantifying location privacy as they are not correlated with the adversary's error in his inference attacks.

**Related Publications** [SFJH09, STD+11, STLBH11]

# 4 Strategic Protection Mechanisms

The disclosure of users' whereabouts to the LBS exposes aspects of their private life that is not apparent at first, but can be inferred from the revealed location data (see Chapter 3 and [FSH11, GP09, Kru07]). A large body of research has focused on developing location-privacy protection mechanisms (LPPMs) that allow users to make use of LBSs and limit the amount of disclosed sensitive information [BS03, CG09, FSH09, GL05, HGXA07, KGMP07, MRC09]. These protection mechanisms are based on hiding or perturbing the real locations of a user, or even sending fake locations to the LBS, in order to increase the uncertainty of the adversary about a user's true whereabouts. However, the evaluation of these designs usually disregards the fact that the adversary might have some knowledge about users' access patterns to the LBS and also about the algorithm implemented by the LPPM. As we show in Chapter 3, such information allows the attacker to reduce his uncertainty on the user's true location. Hence, prior evaluations overestimate the location privacy offered by a protection system.

In this chapter, we focus on a broad range of LBSs and location sharing services in which users reveal their location in a *sporadic* manner, e.g., via applications for location check-in and location-tagging, or via applications for finding nearby points-of-interests, local events, or nearby friends. We consider an adversary interested in uncovering the location of a user at the time she sends the LBS query (i.e., an adversary performing localization attacks). We focus on *user-centric* LPPMs in which the decisions taken to protect privacy (e.g., hiding, perturbing, or faking the location) are made locally by the user. Hence, these LPPMs can be easily integrated in the mobile device that she uses to access LBS. We note that the principles behind our protection mechanism design are applicable to LBSs where users reveal their location continuously (rather than sporadically), and where the adversary's aim is to track users continuously over space and time.

We propose an analytical framework that allows system designers to find the optimal LPPM against a strategic adversary who, knowing each user's LBS access pattern and the underlying obfuscation algorithm, employs an optimal attack to localize them. The challenge is to design an optimal protection mechanism when the inference attack, dependent on the mechanism being designed, is unknown to the designer. As opposed to making any assumption about the adversary's inference algorithm (i.e., limiting his power), we co-infer the optimal attack while finding the defense mechanism. Additionally, our methodology constrains the search space to the LPPMs that obfuscate locations in such a way that the quality of the LBS response is not degraded below a threshold imposed by the user, hence required service quality for the user is guaranteed. We assume that the adversary is also aware of this user-specified service quality constraint. To the best of our knowledge, this is the first analytical framework that allows engineers to methodologically integrate adversarial knowledge in the design of optimal user-centric privacy protection mechanisms.

We formalize the problem of finding the optimal LPPM anticipating the optimal inference attack as an instance of a zero-sum Bayesian Stackelberg game. In this game, a leader and

a follower interact strategically, with each one's gain being the loss of the other. The leader decides on her strategy knowing that it will be observed by the follower, who will optimize his choice based on this observation. In our scenario, the user is the leader and the adversary is the follower. Then, this game precisely models that the adversary knows the user's choice of protection mechanism and will use this knowledge to improve the effectiveness of his attack. We extend the classic formulation of a Stackelberg game with an additional constraint to ensure that the service quality is satisfactory for the user. This enables us to find the optimal point in the tradeoff curve between privacy and service quality, which satisfies both user privacy and service quality requirements.

Our solution is similar to previous work on security in which, as in the location-privacy scenario, the defender can be modeled as a Stackelberg game leader, and the adversary as the follower. The common theme is that the defender must commit to a defense strategy/protocol, which is then disclosed to the adversary, who can then choose an optimal course of action *after* observing the defender's strategy. Paruchuri *et al.* [PPM+08] propose an efficient algorithm for finding the leader's optimal strategy considering as a main case study a patrolling agent who searches for a robber in a limited area. In their case, the defender is unsure about the type of the adversary (i.e. where the adversary will attack). In contrast, in our work it is the adversary who is unsure about the type (i.e. the true location) of the user/defender. A similar approach is used by Liu and Chawla [LC09] in the design of an optimal e-mail spam filter, taking into account that spammers adapt their e-mails to get past the spam detectors. The same problem is tackled by Brückner and Scheffer [BS11], who further compare the Stackelberg-based approach with previous spam filters based on support vector machines, logistic regression, and Nash-logistic regression. Korzhyk *et al.* [KYK+11] contrast the Stackelberg framework with the more traditional Nash framework, within a class of security games. A recent survey [MZA+11] explores the connections between security solutions and game theory. To the best of our knowledge, our work is the first that uses Bayesian Stackelberg games to design optimal privacy-protection mechanisms.

We evaluate the LPPMs generated by our method by using real location traces. We show how, for a given user's LBS access pattern and service-quality threshold, our game-theoretic approach enables us to simultaneously find the optimal LPPM and the optimal attack against it. We confirm that there is a trade-off between the maximum achievable privacy and the service quality, but once a certain privacy level is reached, loosening the quality requirements further does not necessarily result in a privacy gain. We also find that the location-privacy gain of using the optimal LPPM, with respect to a suboptimal one, is larger when the quality constraint is tighter (compared to the case where users' quality requirements allow the LPPM to significantly perturb locations before sending them to the LBS).

## 4.1 The Problem Statement

In this section, we first explain our system model and assumptions, and then sketch the problem that we solve. In Table 4.1, we summarize the notations introduced throughout the section. Note that, as we focus on user-centric mechanisms, which give protection to each user separately, throughout this chapter we omit the user identity $u$.

| | |
|---|---|
| $\boldsymbol{\psi}$ | LBS access profile of the user. Element $\psi(r)$ is the probability of being at location $r$ when accessing the LBS. |
| $f(\tilde{r}\|r)$ | Location obfuscation function implemented by the LPPM: Probability of replacing $r$ with $\tilde{r}$. |
| $h(\hat{r}\|\tilde{r})$ | Adversary's attack function: Probability of estimating $\hat{r}$ as user's actual location, if pseudolocation $\tilde{r}$ is observed. |
| $d_q(\tilde{r}, r)$ | Incurred service-quality loss by the user if LPPM replaces location $r$ with pseudolocation $\tilde{r}$. |
| $d_p(\hat{r}, r)$ | Distance between locations $\hat{r}$ and $r$: Privacy of the user at location $r$ if adversary's estimate is $\hat{r}$. |
| $Q_{loss}(\boldsymbol{\psi}, f, d_q)$ | Expected quality loss of an LPPM with location obfuscation function $f$. |
| $Q_{loss}^{\max}$ | Maximum tolerable service quality loss, determined by the user. |
| $Privacy(\boldsymbol{\psi}, f, h, d_p)$ | Expected location privacy of the user with profile $\boldsymbol{\psi}$ using protection $f$ against attack $h$. |

Table 4.1: Table of notations.

### 4.1.1 User and Adversary

We assume that users connect *sporadically* to an LBS provider to which they need to reveal their current location in order to obtain a service, i.e., there is a non-negligible time gap between two successive accesses of a given user to the LBS. The *access profile* $\boldsymbol{\psi}$ of the user is the probability distribution of the location $r$ from which the user accesses the LBS. More formally, $\psi(r) = \Pr\{\boldsymbol{A}^t = r \,|\, \boldsymbol{X}^t = 1\}$.[1] We assume that the adversary has an estimation of each user's access profile. So, he knows the frequency with which the user issues queries from regions in $\mathcal{R}$, i.e., $\boldsymbol{\psi}$.

### 4.1.2 Location-Privacy Protection Mechanism

We consider that users want to preserve their location privacy when they access the LBS. Users implement a local and user-centric LPPM that transforms each true location $r$ into a pseudolocation $\tilde{r} \in \widetilde{\mathcal{R}}$, which is then sent to the LBS instead of the actual location. For each actual location $r$, the LPPM chooses a pseudolocation $\tilde{r}$ by sampling from the following probability distribution:

$$f(\tilde{r}|r) = \Pr\{\boldsymbol{O}^t = \tilde{r} \,|\, \boldsymbol{A}^t = r, \boldsymbol{X}^t = 1\} \tag{4.1}$$

As accesses to the LBS are sporadic, two successive query locations of the user are *conditionally* independent given $\boldsymbol{\psi}$. The larger the inter-query time is, the more independent the two locations of the user in her successive LBS accesses are. This is also reflected in the LPPM's obfuscation algorithm that outputs pseudolocations that depend only on the user's current location.

[1] We note that, similar to the mobility profile, the access profile is also time-dependent (e.g., users have different access patterns in the morning than in the afternoon). This dependency also affects users' location privacy requirements, and service quality requirements. For the sake of simplicity, we omit the time-period and provide a solution for each user in a given time period. But, we note that the method is easily adaptable to more complex access patterns and privacy/quality requirements elicitation that account for such changes in time (e.g., by applying the method to each time period separately).

### 4.1.3 Service Quality Metric

In the aforementioned setting, the LBS response quality depends on the pseudolocation output by the LPPM and not on the user's actual location. The distortion introduced in the observed pseudolocations determines the quality of service that the user experiences. The more similar the actual and the observed location are, the higher the service quality is. We compute the expected *quality loss* due to an LPPM $f(.)$ as an average of $d_q(\tilde{r}, r)$ over all $r$ and $\tilde{r}$:

$$Q_{loss}(\boldsymbol{\psi}, f, d_q) = \sum_{r,\tilde{r}} \psi(r) \cdot f(\tilde{r}|r) \cdot d_q(\tilde{r}, r). \tag{4.2}$$

Function $d_q(.)$ determines the dissimilarity between location $r$ and pseudolocation $\tilde{r}$. The semantics of this dissimilarity depend on the LBS under consideration and also on the user's specific service-quality expectations. In many applications, the service quality can be considered inversely proportional to the physical distance between $r$ and $\tilde{r}$. For example, applications that find nearby points of interest can give very different responses to $r$ and to $\tilde{r}$ even if they are only a couple of kilometers apart. In contrast, there exist LBSs in which the service quality depends on other criteria, such as on whether $\tilde{r}$ is within a region of interest. For a weather forecast application, for instance, any pseudolocation $\tilde{r}$ in the same city as the actual location $r$ would result in a high quality LBS response.

We assume that each user imposes a maximum tolerable service quality loss, $Q_{loss}^{\max}$, caused by sharing pseudolocations instead of their actual locations. Formally,

$$Q_{loss}(\boldsymbol{\psi}, f, d_q) \leq Q_{loss}^{\max}. \tag{4.3}$$

This constrains the LPPM obfuscation function $f(\tilde{r}|r)$ to not output pseudolocations that, on average, would result in lower quality. We note that the influence of threshold $Q_{loss}^{\max}$ on the LPPM depends on the function $d_q(.)$, hence it is also dependent on the type of the LBS the user is querying. In the case of an LBS that finds nearby points of interest, where $d_q(.)$ is proportional to the physical distance between $r$ and $\tilde{r}$, enforcing the quality threshold results in ensuring a maximum expected distance between these two locations. For the weather application, enforcing the quality threshold results in setting region boundaries within which locations lead to the same forecast. For other location-based applications, the function $d_q(.)$ and the threshold $Q_{loss}^{\max}$ is defined in a similar way.

### 4.1.4 Location Privacy Metric

The adversary's goal is to infer the user's actual events $\boldsymbol{A}^t = r$, given the observed events $\boldsymbol{O}^t = \tilde{r}$. Recall that the adversary knows the user's profile, $\boldsymbol{\psi}$. He uses this background knowledge to run an inference attack on the observed events in order to output estimations $\hat{r}$ of the user's actual locations. Formally, the attack result can be described as a probability density function $h(.)$ such that

$$h(\hat{r}|\tilde{r}) = \Pr\{\boldsymbol{A}^t = \hat{r} \,|\, \boldsymbol{O}^t = \tilde{r}\}. \tag{4.4}$$

Under the sporadic assumption about the LBS, the current (query) location of the user is conditionally independent of her past and future observed locations, given her access profile. This is reflected in that the computation of the estimated location $\hat{r}$ at time $t$ only depends on the pseudolocation $\tilde{r}$ observed at the same time $t$.

We note that the attack formulation is independent of whether the considered LPPM anonymizes the events or not. In this work, we assume that the adversary knows the identity of the users behind the events, but the framework can be adapted to anonymous LPPMs as well. Note that even when users are anonymous, our optimal solution provides a guarantee for their location privacy (even after a potential re-identification attack).

We quantify the user's location privacy as the adversary's expected error in his inference attack (see Section 1.5.1), i.e., the expected distortion in the reconstructed event. We compute the expectation over all $r, \tilde{r}$, and $\hat{r}$:

$$Privacy(\boldsymbol{\psi}, f, h, d_p) = \sum_{\hat{r},\tilde{r},r} \psi(r) \cdot f(\tilde{r}|r) \cdot h(\hat{r}|\tilde{r}) \cdot d_p(\hat{r}, r) \tag{4.5}$$

The distortion function quantifies the loss of privacy stemming from the inference attack. The privacy loss depends on the locations' semantics and also on the privacy requirements of the user (i.e., some users might consider locations inside a hospital more sensitive than other places), and $d_p(.)$ must be defined accordingly. For instance, if the user wants to hide just her exact current location (as opposed to hiding her location area), the appropriate distortion function could be the Hamming distance (probability of error) between the estimated location $\hat{r}$ and the actual location $r$:

$$d_p(\hat{r}, r) = \begin{cases} 0, & \text{if } \hat{r} = r \\ 1, & \text{otherwise} \end{cases} \tag{4.6}$$

In this case, any location different from the user's actual location results in a high level of location privacy. Alternatively, the user's privacy might depend on the physical distance between the estimated and actual locations, hence the distortion function can be modeled as the Euclidean distance between these locations, i.e., the squared-error distortion:

$$d_p(\hat{r}, r) = (\hat{r} - r)^2 \tag{4.7}$$

### 4.1.5 Problem Statement

Given

1. a maximum tolerable service-quality loss $Q_{loss}^{\max}$ imposed by the user as a bound for $Q_{loss}(.)$, computed using the quality function $d_q(.)$, and
2. a prior adversarial knowledge of the user's profile $\boldsymbol{\psi}$,

the problem is finding the LPPM obfuscation function $f(.)$ that maximizes the user's location privacy as defined in (4.5). The solution must consider that the adversary

1. observes the LPPM's output $\tilde{r}$, and
2. is aware of the LPPM's internal algorithm $f(.)$.

Hence, the adversary implements the *optimal* attack $h(.)$ that estimates the true location of the user with the least distortion as measured by $d_p(.)$.

## 4.2 Game Formulation

The problem of finding an LPPM that offers optimal location privacy given the knowledge of the adversary is an instance of a zero-sum Bayesian Stackelberg game. In a Stackelberg game the *leader*, in our case the user, plays first by choosing an LPPM and committing to it by running it on her actual location. The *follower*, in our case the adversary, plays next by estimating the user's location, knowing the LPPM that the user has committed to. It is a Bayesian game because the adversary has incomplete information about the user's true location and plays according to his hypothesis about this location. It is also an instance of a zero-sum game, as the adversary's gain (or loss) of utility is exactly balanced by the losses (or gains) of the utility of the user: the information gained (lost) by the adversary is the location privacy lost (gained) by the user. We now proceed to define the game adapted to our problem:

**Step 0** Nature selects a location $r \in \mathcal{R}$ for the user to access the LBS, according to a probability distribution $\boldsymbol{\psi}$. That is, location $r$ is selected with probability $\psi(r)$.

**Step 1** Given $r$, the user runs the LPPM $f(\tilde{r}|r)$ to select a pseudolocation $\tilde{r} \in \widetilde{\mathcal{R}}$, subject to $f(.)$ complying with the service quality constraint (4.3).

**Step 2** Having observed $\tilde{r}$, the adversary selects an estimated location $\hat{r} \sim h(\hat{r}|\tilde{r}), \hat{r} \in \mathcal{R}$. The adversary knows the probability distribution $f(\tilde{r}|r)$ used by the LPPM; he also knows the user's profile $\boldsymbol{\psi}$, but not the true location $r$.

**Final Step** The adversary pays an amount $d_p(\hat{r}, r)$ to the user. This amount represents the adversary's error (equivalently, the location privacy gained by the user).

The above description is common knowledge to both the adversary and the user. They both aim to maximize their payoff, i.e. the adversary tries to minimize the expected amount that he will pay, whereas the user tries to maximize it.

## 4.3 Solution

In this section, we describe a precise optimization problem that formalizes the objectives of the user and of the adversary. We construct two linear programs that, given $\boldsymbol{\psi}$, $d_p(.)$ and $d_q(.)$, we compute the user's optimal choice of protection mechanism $f(.)$, and the adversary's optimal choice of inference attack $h(.)$.

### 4.3.1 Optimal Strategy for the User

The adversary observes the pseudolocation $\tilde{r}$ output by the LPPM, he knows the function $f(\tilde{r}|r)$ implemented by the LPPM, and he also knows the user's access profile $\boldsymbol{\psi}$. Thus, he can form the posterior distribution

$$\Pr\{r|\tilde{r}\} = \frac{\mathbb{P}\mathrm{r}\{r, \tilde{r}\}}{\mathbb{P}\mathrm{r}\{\tilde{r}\}} = \frac{f(\tilde{r}|r) \cdot \psi(r)}{\sum_{r'} f(\tilde{r}|r') \cdot \psi(r')} \tag{4.8}$$

on the true location $r$ of the user, conditional on the observation $\tilde{r}$. The adversary's objective is then to choose $\hat{r}$ to minimize the user's conditional expected privacy, where the expectation

is taken under $\mathbb{P}\mathrm{r}\{r|\tilde{r}\}$. The user's conditional expected privacy for an arbitrary $\hat{r}$ is

$$\sum_r \mathbb{P}\mathrm{r}\{r|\tilde{r}\} \cdot d_p(\hat{r}, r), \tag{4.9}$$

and for the minimizing $\hat{r}$ it is

$$\min_{\hat{r}} \sum_r \mathbb{P}\mathrm{r}\{r|\tilde{r}\} \cdot d_p(\hat{r}, r). \tag{4.10}$$

If there are multiple values of $\hat{r}$ that satisfy (4.10), then the adversary randomizes arbitrarily among them. The probability with which $\hat{r}$ is chosen in this randomization is $h(\hat{r}|\tilde{r})$. Of course, $h(\hat{r}|\tilde{r})$ will be positive only for minimizing values of $\hat{r}$; for all other values $h(\hat{r}|\tilde{r})$ will be zero. When randomizing, (4.10) is rewritten as

$$\sum_{r,\hat{r}} \mathbb{P}\mathrm{r}\{r|\tilde{r}\} \cdot h(\hat{r}|\tilde{r}) \cdot d_p(\hat{r}, r). \tag{4.11}$$

Note that if there is only one value of $\hat{r}$ satisfying (4.10), then this value is selected with probability 1 in the randomization, whereas all other values are selected with probability 0, so (4.11) reduces to (4.10). In this sense, (4.11) is a generalization of (4.10), but it should be noted that both expressions compute the same conditional expected privacy.

We see that for a given $\tilde{r}$, the user's conditional privacy is given by (4.10). The probability that $\tilde{r}$ is output by the LPPM is $\mathbb{P}\mathrm{r}\{\tilde{r}\} = \sum_r f(\tilde{r}|r) \cdot \psi(r)$. Hence, the user's *unconditional* expected privacy (averaged over all $\tilde{r}$) is

$$\sum_{\tilde{r}} \mathbb{P}\mathrm{r}\{\tilde{r}\} \min_{\hat{r}} \sum_r \mathrm{Pr}(r|\tilde{r}) \cdot d_p(\hat{r}, r) = \sum_{\tilde{r}} \min_{\hat{r}} \sum_r \psi(r) \cdot f(\tilde{r}|r) \cdot d_p(\hat{r}, r). \tag{4.12}$$

To facilitate the computations, we define

$$x_{\tilde{r}} \triangleq \min_{\hat{r}} \sum_r \psi(r) \cdot f(\tilde{r}|r) \cdot d_p(\hat{r}, r). \tag{4.13}$$

Incorporating $x_{\tilde{r}}$ into (4.12), we rewrite the unconditional expected privacy of the user as

$$\sum_{\tilde{r}} x_{\tilde{r}}, \tag{4.14}$$

which the user aims to maximize by choosing the optimal $f(\tilde{r}|r)$. The minimum operator makes the problem non-linear, which is undesirable, but (4.13) can be transformed to a series of linear constraints:

$$x_{\tilde{r}} \leq \sum_r \psi(r) \cdot f(\tilde{r}|r) \cdot d_p(\hat{r}, r), \ \forall \hat{r}. \tag{4.15}$$

It turns out that maximizing (4.14) under (4.13) is equivalent to maximizing (4.14) under (4.15) [DPV08, Ch. 7, p. 224].

We construct the linear program for the user from (4.14) and (4.15). Note that variable $x_{\tilde{r}}$ is a *decision* variable in the linear program, i.e. it is among the quantities chosen by the solver. This might appear counterintuitive, as $x_{\tilde{r}}$ is defined in (4.13) as a function of

$f(.)$, rather than as an independent variable that can be freely selected. But, because of the transformation, it is always guaranteed that (4.13) will hold.

The linear program for the user is the following: Choose $f(\tilde{r}|r), x_{\tilde{r}}, \forall r, \tilde{r}$ in order to

$$\textbf{Maximize } \sum_{\tilde{r}} x_{\tilde{r}} \tag{4.16}$$

**subject to**

$$x_{\tilde{r}} \leq \sum_{r} \psi(r) \cdot f(\tilde{r}|r) \cdot d_p(\hat{r}, r), \ \forall \hat{r}, \tilde{r} \tag{4.17}$$

$$\sum_{r} \psi(r) \sum_{\tilde{r}} f(\tilde{r}|r) \cdot d_q(\tilde{r}, r) \leq Q_{loss}^{\max} \tag{4.18}$$

$$\sum_{\tilde{r}} f(\tilde{r}|r) = 1, \ \forall r \tag{4.19}$$

$$f(\tilde{r}|r) \geq 0, \ \forall r, \tilde{r} \tag{4.20}$$

Inequalities (4.17) are the series of linear constraints (4.15), one series for each value of $r'$; inequality (4.18) reflects the service quality constraint; constraints (4.19) and (4.20) reflect that $f(\tilde{r}|r)$ is a probability distribution function.

### 4.3.2 Optimal Strategy for the Adversary

The reasoning is similar for the formalization of the adversary's optimization problem. When the LPPM's output is pseudolocation $\tilde{r}$, the adversary will solve (4.10) to find an estimate $\hat{r}$. In general, the adversary will find multiple minimizing values of $\hat{r}$, and each of them will be selected with some probability $h(\hat{r}|\tilde{r})$. Given that the true location is $r$ and that the observed pseudolocation is $\tilde{r}$, the conditional expected user privacy is

$$\sum_{\hat{r}} h(\hat{r}|\tilde{r}) \cdot d_p(\hat{r}, r). \tag{4.21}$$

In her optimal strategy, the user chooses $\tilde{r}$ to maximize (4.21). So, given that the true location is $r$, the conditional expected user privacy for the maximizing $\tilde{r}$ is

$$y_r \triangleq \max_{\tilde{r}} \sum_{\hat{r}} h(\hat{r}|\tilde{r}) \cdot d_p(\hat{r}, r). \tag{4.22}$$

Similarly as before, the maximization can be generalized to a randomization among maximizing values of $\tilde{r}$. The probability with which $\tilde{r}$ is chosen is $f(\tilde{r}|r)$.

The prior distribution $\psi(r)$ contains the adversary's knowledge of $r$. Thus, the unconditional expected user privacy is

$$\sum_{r} \psi(r) \cdot y_r, \tag{4.23}$$

that the adversary aims to minimize by choosing $h(\hat{r}|\tilde{r})$. Similarly as before, (4.22) can be transformed to an equivalent series of linear constraints:

$$y_r \geq \sum_{\hat{r}} h(\hat{r}|\tilde{r}) \cdot d_p(\hat{r}, r), \forall \tilde{r}. \tag{4.24}$$

We construct the linear program for the adversary (which is the dual of the user's linear program) from (4.23) and (4.24): Choose $h(\hat{r}|\tilde{r}), y_r, \forall r, \tilde{r}, \hat{r}$, and $z \in [0, \infty)$ in order to

$$\textbf{Minimize} \quad \sum_r \psi(r) \cdot y_r + z \cdot Q_{loss}^{\max} \tag{4.25}$$

**subject to**

$$y_r \geq \sum_{\hat{r}} h(\hat{r}|\tilde{r}) \cdot d_p(\hat{r}, r) + z \cdot d_q(\tilde{r}, r), \forall r, \tilde{r} \tag{4.26}$$

$$\sum_{\hat{r}} h(\hat{r}|\tilde{r}) = 1, \forall \tilde{r} \tag{4.27}$$

$$h(\hat{r}|\tilde{r}) \geq 0, \forall \tilde{r}, \hat{r} \tag{4.28}$$

$$z \geq 0 \tag{4.29}$$

Note the role of variable $z$: In linear programming parlance, it is the *shadow price* of the service quality constraint. Intuitively, $z$ is the "exchange rate" between service quality and privacy. Its value in the optimal solution indicates the amount of privacy (in privacy units) that is lost (gained) if the service quality threshold $Q_{loss}^{\max}$ increases (decreases) by one unit of quality.

For example, if $z > 0$ in the optimal solution, then any change $\Delta Q_{loss}^{\max}$ in $Q_{loss}^{\max}$ will affect the privacy achieved by $z \cdot \Delta Q_{loss}^{\max}$. In this case, constraint (4.18) is satisfied as a strict equality. In contrast, if constraint (4.18) is satisfied as a strict inequality, then, intuitively, the selection of $f(\tilde{r}|r)$ has not been constrained by $Q_{loss}^{\max}$. In this case, any (small) changes in $Q_{loss}^{\max}$ will not have any effect on $f(\tilde{r}|r)$, nor on the privacy achieved. So, $z$ would be zero.

Note that both linear programs (to obtain optimal strategies for the user and the adversary) compute the unconditional expected privacy of the user (4.5). Previous expressions can be derived from this one. For instance, if there is a single best choice of a pseudolocation $\tilde{r}$ for each given location $r$, then $f(\tilde{r}|r)$ is always either 0 or 1, so (4.10) is obtained. The optimal solution of each linear program results in the same value for the privacy of the user. Hence, in principle, we only need to compute one of the two to quantify maximum level of privacy of the user. We choose to present both, because the user's linear program incorporates the service quality constraint in a more straightforward manner, whereas the adversary's linear program explicitly computes the "exchange rate" between service quality and privacy.

## 4.4 Evaluation

The proposed optimization framework enables us to determine the most effective location-privacy preserving mechanism (LPPM) against optimal inference attacks. The optimal LPPM is designed under the constraint of guaranteeing a minimum service quality such that the location-based service remains useful for the user. In this section, we evaluate the relation between location privacy and service quality for a few example location-based services (Recall that the service-quality sensitivity of a LBS to location obfuscation is encoded through the dissimilarity function $d_q(.)$). Moreover, we evaluate the performance of non-optimal LPPMs and non-optimal inference attacks against the optimal strategies.

We use real location traces of people (in Lausanne, Switzerland) who use various means of

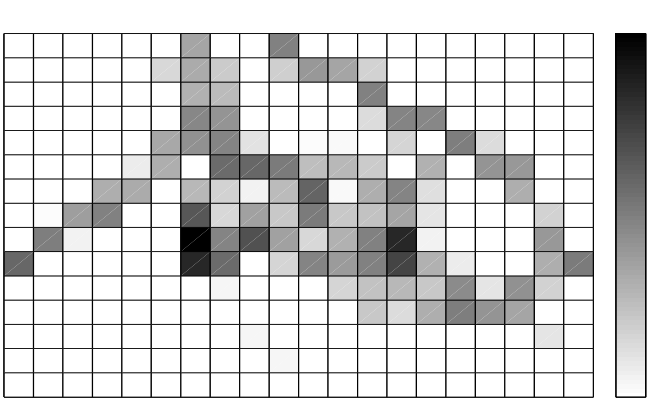


Figure 4.1: Spatial histogram showing the density of users per region (in log scale) in Lausanne. The area size is 15.32km × 7.58km, divided into 20 × 15 regions.

transportation.[2] We select 11 users at random, and we focus on their location traces during the day (8am to 8pm), when it is more probable that users use location-based services. The duration of the considered traces is one month. The location area, within which they move, is divided into 300 regions. Figure 4.1 shows the density of users across all the regions. The grayness of the cells shows the density of its corresponding region in log scale. As many of the regions are not (or very rarely) visited by many individual users, we compute each user's profile $\boldsymbol{\psi}$ by considering only the 30 most popular regions across the whole population. This prevents sparse user profiles. A user's profile is the normalized number of her visits to each region.

Given distance functions $d_p(.)$ and $d_q(.)$ and service-quality loss threshold $Q_{loss}^{\max}$, we compute the optimal LPPM and its corresponding optimal attack by solving (4.16) and (4.25) using Matlab's linear programming solver. We then compare the obtained optimal protection mechanism and the optimal inference attack against obfuscation LPPMs and Bayesian inference attacks, respectively.

### 4.4.1 Basic Obfuscation LPPM

The basic obfuscation LPPM, with an obfuscation level $k = 1, 2, 3, \ldots$, is constructed in the following way: For each location $r$, we find its $k-1$ closest locations (using the Euclidean distance between the centers of the regions). The probability distribution function $f(.|r)$ will be the uniform probability distribution on the set of the $k-1$ selected locations together with the location $r$. That is, location $r$ is replaced by each of the $k$ locations, as a pseudolocation, with the same probability $\frac{1}{k}$, and all the rest of locations have probability 0. Thus, in practice, an actual location $r$ is hidden among its $k-1$ nearest locations. We choose this mechanism, as it is very popular in the literature.

Given the user profile $\boldsymbol{\psi}$ and quality distance function $d_q(.)$, we use (4.2) to compute the expected service-quality loss $Q_{loss}(\boldsymbol{\psi}, f, d_q)$ for any LPPM obfuscation $f(.)$, whether it be optimal or not.

[2] The traces are obtained from the Lausanne Data Collection Campaign dataset, http://research.nokia.com/page/11367

### 4.4.2 Bayesian Inference Attack on an LPPM

We compare the effectiveness of our optimal attack with the Bayesian inference attack, which has been shown effective in Chapter 3. In the Bayesian approach, for each pseudolocation $\tilde{r}$, the posterior probability distribution over the locations is used to invert the noise added by the LPPM. The posterior probability is computed as

$$h(\hat{r}|\tilde{r}) = \frac{\Pr\{\hat{r}, \tilde{r}\}}{\Pr\{\tilde{r}\}} = \frac{f(\tilde{r}|\hat{r}) \cdot \psi(\hat{r})}{\sum_{r'} f(\tilde{r}|r') \cdot \psi(r')}. \tag{4.30}$$

We use (4.5) to compute the expected location privacy of a user who adopts a given (obfuscation or optimal) LPPM $f(.)$ against a (Bayesian or optimal) inference attack $h(.)$. The expected location privacy also depends on the distortion function $d_p(.)$ that we choose to use.

Briefly, if $d_p(.)$ is the Hamming distance, then the Bayesian attack chooses the location with the highest posterior probability $\Pr\{\hat{r}|\tilde{r}\}$. If $d_p(.)$ is the Euclidean distance, the Bayesian attack chooses the conditional expected value $\mathbb{E}\{\hat{r}|\tilde{r}\}$.

### 4.4.3 Optimal Inference Attack on an Arbitrary LPPM

In order to make a fair comparison between the effectiveness of the optimal and obfuscation LPPM, we need to run the same attack on both of them. The Bayesian inference attack described by (4.30) can be performed against both. However, we still need to design an optimal attack against arbitrary LPPMs that have not been constructed in our game-theoretic framework.

The optimal inference attack is the one that minimizes the expected user privacy:

$$h(\hat{r}|\tilde{r}) = \arg\min_{h} Privacy(\boldsymbol{\psi}, f, h, d_p). \tag{4.31}$$

Given the user profile $\boldsymbol{\psi}$, an LPPM $f(.)$ and distortion function $d_p(.)$, the following linear program finds the optimal attack $h(.)$. Note that, compared to (4.25), there is no service quality constraint here, as the LPPM has been assumed to be arbitrary.

$$\textbf{Minimize} \sum_{\hat{r},\tilde{r},r} \psi(r) \cdot f(\tilde{r}|r) \cdot h(\hat{r}|\tilde{r}) \cdot d_p(\hat{r}, r) \tag{4.32}$$

$$\textbf{subject to} \sum_{\hat{r}} h(\hat{r}|\tilde{r}) = 1, \forall \tilde{r}, \text{and } h(\hat{r}|\tilde{r}) \geq 0, \forall \hat{r}, \tilde{r} \tag{4.33}$$

### 4.4.4 Location-Privacy Protection Mechanism Output

Consider a LBS user making use of our optimal LPPM on her mobile device. The way her location appears in the eyes of the adversary is shown in Figure 4.2. For the sake of comparison, Figure 4.2 also shows how a basic obfuscation LPPM distributes the pseudolocations over space. In order to make a fair comparison, we need to make sure that the cost of the two LPPMs, in terms of service quality, is the same. To do so, we compute the quality loss $Q_{loss}$ of the obfuscation LPPM and assign this loss as the quality threshold $Q_{loss}^{\max}$ of the optimal LPPM. Hence, the optimal LPPM cannot sacrifice the service quality more than the obfuscation LPPM to gain higher location privacy.

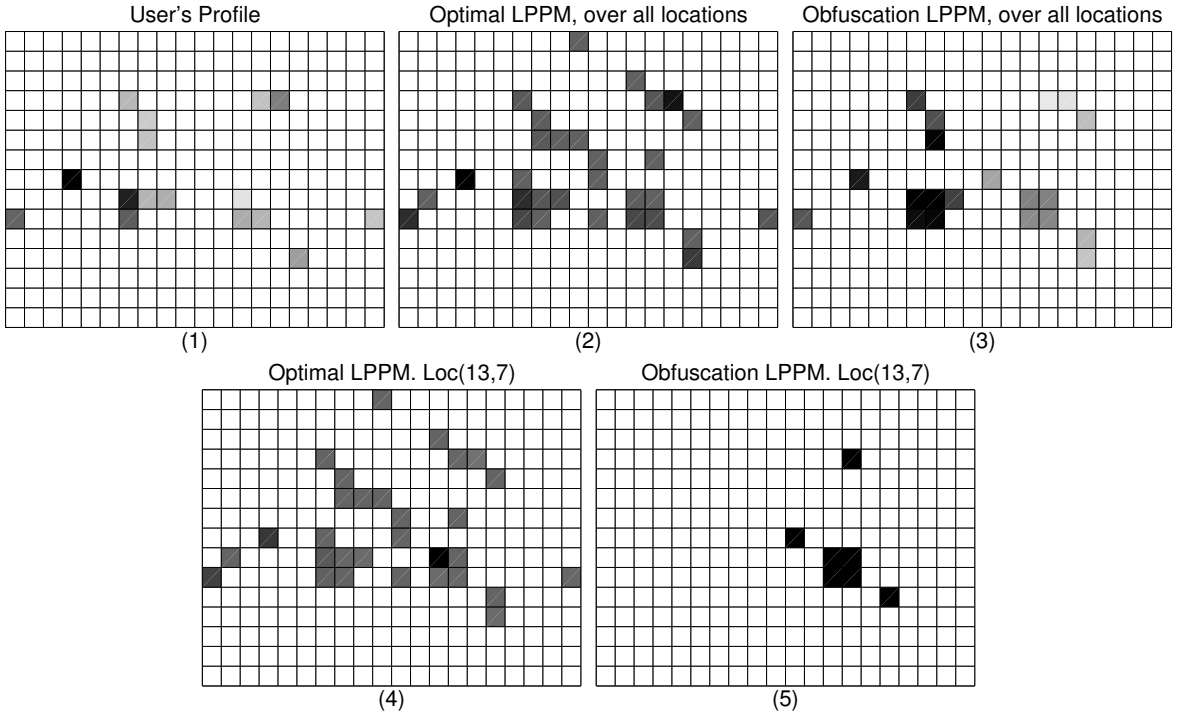


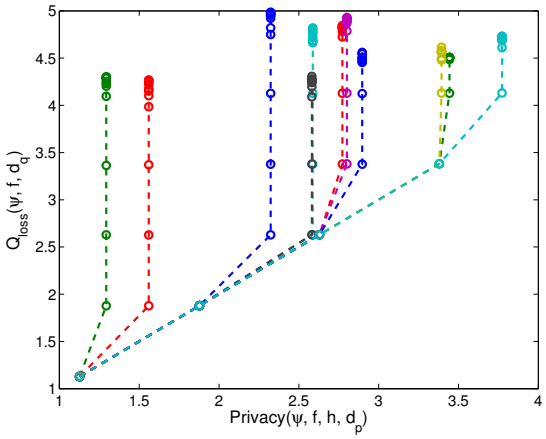

Figure 4.2: Input/Output of LPPM. Profile of a user for whom the subsequent calculations are made (sub-figure 1). Distribution $\mathbb{P}\mathrm{r}\{\tilde{r}\}$ of observed pseudolocations when using the optimal LPPM with $Q_{loss}^{\max} = 0.8690$ (sub-figure 2). Distribution $\mathbb{P}\mathrm{r}\{\tilde{r}\}$ of observed pseudolocations when using obfuscation LPPM with $Q_{loss}(\boldsymbol{\psi}, f, d_q) = 0.8690$ (sub-figure 3). Conditional distribution $\mathbb{P}\mathrm{r}\{\tilde{r}|r\}$ when using the optimal LPPM on location $r = (13, 7)$ (sub-figure 4). Conditional distribution $\mathbb{P}\mathrm{r}\{\tilde{r}|r\}$ when using obfuscation LPPM on location $r = (13, 7)$ (sub-figure 5). In the grid, column 1 is the left-most column, and row 1 is the bottom row. We assume Euclidean distortion $d_p$ and Hamming distortion $d_q$.

Figures 4.2(2) and 4.2(3) show $\mathbb{P}\mathrm{r}\{\tilde{r}\}$, the distribution of pseudolocations averaged over all locations for optimal and basic obfuscation LPPMs, respectively. Given arbitrary LPPM location obfuscation function $f(.)$ and user profile $\boldsymbol{\psi}$, the probability distribution of pseudolocations is

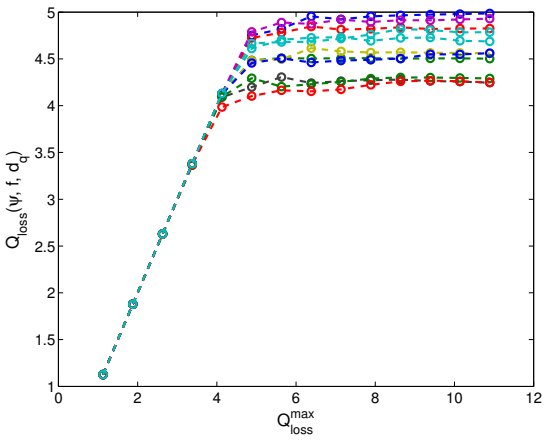
$$\mathbb{P}\mathrm{r}\{\tilde{r}\} = \sum_{r} \psi(r) \cdot f(\tilde{r}|r). \tag{4.34}$$

As it is shown, the distribution corresponding to the optimal LPPM is more uniform, making it more difficult for the adversary to invert it effectively.

In Figures 4.2(4) and 4.2(5), we show the distribution of pseudolocations for specific location $r = loc(13, 7)$. By observing how uniform their outputs are, we can easily make the comparison between the two LPPMs. The obfuscation LPPM is obviously more concentrated around the actual location, whereas the optimal LPPM (with the same service-quality loss as the obfuscation method) broadens the set of pseudolocations to most of possible regions including highly probable regions (i.e. regions $r$ with a large $\psi(r)$). This higher diversity brings higher privacy, as we will see later in this section.




(a) Location privacy $Privacy(\boldsymbol{\psi}, f, h, d_p)$ vs. Service-quality loss $Q_{loss}(\boldsymbol{\psi}, f, d_q)$ for a given service-quality threshold $Q_{loss}^{\max}$. The circles ∘ represent different values of $Q_{loss}^{\max}$.




(b) Service-quality threshold $Q_{loss}^{\max}$ vs. Service-quality loss $Q_{loss}(\boldsymbol{\psi}, f, d_q)$, for a given level of location privacy $Privacy(\boldsymbol{\psi}, f, h, d_p)$. The circles ∘ represent different values of $Privacy(\boldsymbol{\psi}, f, h, d_p)$.

Figure 4.3: Tradeoff between Privacy and Service Quality: Optimal LPPM against the optimal attack. The different lines represent users with diverse profiles $\boldsymbol{\psi}$. We assume Euclidean distortion $d_p$ and Euclidean distortion $d_q$.

### 4.4.5 Tradeoff between Privacy and Service Quality

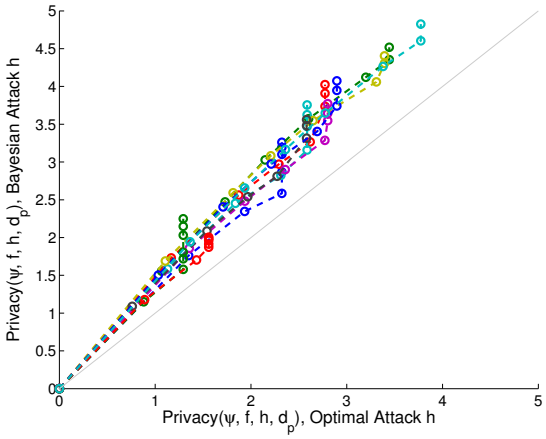
We now study the tradeoff between the level of privacy that the optimal LPPM provides, against the optimal attack, and the service-quality loss that it causes. We plot in Figure 4.3(a) the evolution of the service quality loss, as the optimal LPPM is configured to guarantee different levels of service quality (for users with diverse profiles and for various service quality thresholds). Each line in the figure represents one user and each ∘ represents one $Q_{loss}^{\max}$. We plot $Privacy(\boldsymbol{\psi}, f, h, d_p)$ versus $Q_{loss}(\boldsymbol{\psi}, f, d_q)$.

Unsurprisingly, increasing the level of location-privacy protection significantly degrades the service quality. Also, as expected, we can observe that the maximum achievable location privacy is strongly dependent on the user profile. This is reflected by the separation between the different lines. Each user can have up to a certain level of privacy, regardless of the quality threshold (represented by ∘ in the figure). Hence, the service-quality loss remains constant once this level has been reached. This is due to the presence of the optimal attack that squeezes the location-privacy gain.

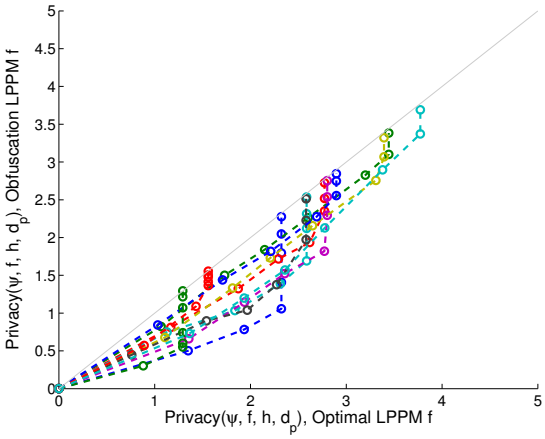
This effect is further illustrated in Figure 4.3(b), where the service-quality loss of optimal LPPM is plotted against the service-quality threshold. Once the optimal LPPM offers the maximal location privacy for a given user profile, loosening the service-quality constraint does not significantly change the LPPM's underlying function $f$, thus there is no reduction in service quality. In other words, it is pointless to sacrifice the service quality, because doing so does not increase the user's location privacy.

### 4.4.6 Effectiveness of the Optimal Strategies

Assuming Euclidean distance functions $d_p(.)$ and $d_q(.)$, we compute the optimal LPPM and attack methods for a set of service quality thresholds $Q_{loss}^{\max}$. For each user, we run the Bayesian




(a) Location privacy $Privacy(\boldsymbol{\psi}, f, h, d_p)$ offered by the optimal LPPM against the optimal attack derived using the game theoretic approach vs. against the Bayesian-inference attack.




(b) Location privacy $Privacy(\boldsymbol{\psi}, f, h, d_p)$ offered by the optimal LPPM vs. location privacy offered by the basic obfuscation LPPM, both evaluated against the optimal attack.

Figure 4.4: Effectiveness of the optimal attack and optimal LPPM strategies. Different lines represent users with diverse profiles $\boldsymbol{\psi}$, and the circles $\circ$ represent different values of $Q_{loss}^{\max}$. We assume Euclidean distortion $d_p$ and Euclidean distortion $d_q$.

inference attack on her optimal LPPM. We also evaluate the location privacy offered by the basic obfuscation LPPM with respect to the optimal attack. We vary the obfuscation level from 1 (minimum) to 30 (maximum), and for each case we compute the corresponding quality loss. Then, this value is set as the threshold $Q_{loss}^{\max}$ for finding the optimal attack mechanism.

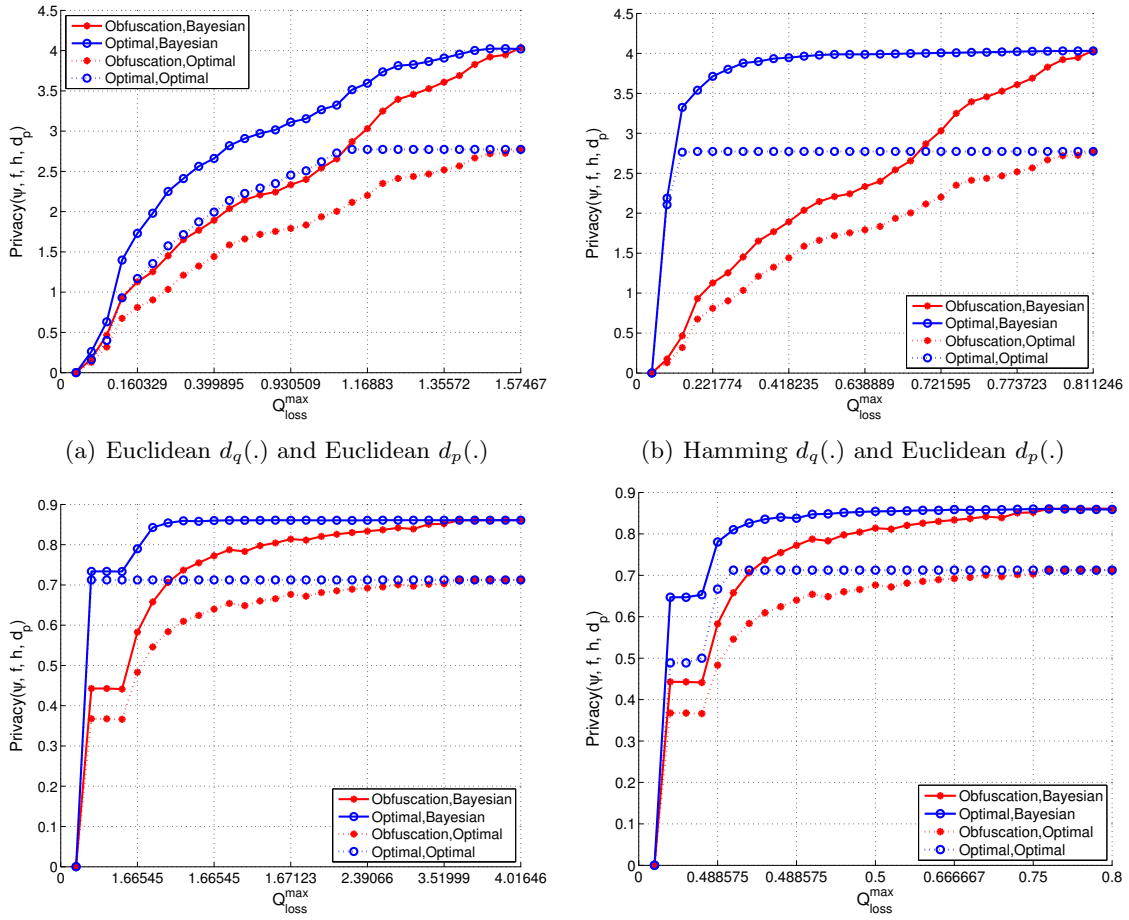
Figure 4.4(a) shows the superiority of the optimal attack to the Bayesian attack, when the location privacy of users is protected by using the optimal LPPM: For any given user and service-quality threshold, the location privacy that the user obtains is smaller when the adversary implements the optimal strategy rather than the Bayesian inference attack.

Figure 4.4(b) shows the superiority of the optimal LPPM to the obfuscation LPPM, against the optimal attack: For any given user and service-quality threshold, a user has a higher privacy level when the LPPM implements the optimal strategy. As expected, the privacy obtained by both mechanisms become equal when no service quality is guaranteed for the user (i.e., $Q_{loss}^{\max}$ is set to its maximum value).

Consider a single user. To further investigate the effectiveness of optimal strategies, we evaluate her privacy under four different combinations of optimal and non-optimal protection/attack methods that have been explained before.

Similar to Figure 4.2, we consider the basic obfuscation LPPM as the basis for generating the service-quality threshold $Q_{loss}^{\max}$. In all graphs of Figure 4.5 each dot represents one obfuscation level used in the basic obfuscation LPPM. The corresponding service-quality loss for each obfuscation level is shown on the x-axis of all four plots. As it can be easily observed from the figures, the optimal attack, compared with the Bayesian attack, always results in a higher degradation of the user's location privacy. Moreover, the optimal LPPM always provides a higher level of privacy for the user (regardless of the service-quality threshold) compared with the basic obfuscation LPPM.

The figures well illustrate how both user and adversary converge to use optimal strategies




(a) Euclidean $d_q(.)$ and Euclidean $d_p(.)$

(b) Hamming $d_q(.)$ and Euclidean $d_p(.)$

(c) Euclidean $d_q(.)$ and Hamming $d_p(.)$

(d) Hamming $d_q(.)$ and Hamming $d_p(.)$

Figure 4.5: Service-quality threshold $Q_{loss}^{\max}$ vs. Location privacy $Privacy(\boldsymbol{\psi}, f, h, d_p)$, for one single user. The different lines represent combinations of optimal ($\circ$) and basic obfuscation ($\bullet$) LPPMs tested against optimal ($\cdots$) and Bayesian-inference ($-$) attacks. The service-quality threshold $Q_{loss}^{\max}$ is equal to the service quality obtained by the basic obfuscation LPPM when the number of obfuscation levels used to perturb the location varies from 1 to 30 (its maximum value).

against each other. The user's favorite setting, i.e. the one that brings her a high level of privacy, is (Optimal, Bayesian). Inversely, the (Obfuscation, Optimal) combination is the favorite setting for the adversary, in which he pays the minimum cost of estimation-error. However, neither of these two settings is a stable state. In the (Optimal, Bayesian) combination, the adversary would gain more by choosing the Optimal attack. In the (Obfuscation, Optimal) combination, the user would gain more by choosing the Optimal LPPM. Hence, the (Optimal, Optimal) combination is a stable equilibrium for both.

The fourth combination (Obfuscation, Bayesian) illustrates an interesting behavior. For small values of quality threshold $Q_{loss}^{\max}$ (or, equivalently, smaller obfuscation levels) the user's privacy is lower than the (Optimal, Optimal) case. However at some middle point, its provided privacy increases and surpasses the privacy obtained from the optimal methods. Indeed,

for small $Q_{loss}^{\max}$, the optimal LPPM uses all its available capacity to increase privacy by distributing the user's pseudolocations over a higher number of locations. So, it performs better than the basic obfuscation LPPM, which is limited to distributing pseudolocations only in a small set of regions. But when the obfuscation level (or, similarly, the service-quality threshold) increases, the basic obfuscation LPPM does better: First, because it is no longer severely limited. And, second, because it is paired against the Bayesian inference attack, which is weaker than the optimal inference attack.

## 4.5 Summary

Accessing location-based services from mobile devices entails a privacy risk for users whose sensitive information can be inferred from the locations they visit. This information leakage raises the need for robust location-privacy protecting mechanisms (LPPMs). In this chapter, we have proposed a game-theoretic framework that enables a designer to find the optimal LPPM for a given location-based service, ensuring a satisfactory service quality for the user. This LPPM is designed to provide user-centric location privacy, hence it is ideal to be implemented in the users' mobile devices.

Our method accounts for the fact that the strongest adversary not only observes the perturbed location sent by the user but also knows the algorithm implemented by the protection mechanism. Hence, he can exploit the information leaked by the LPPM's algorithm to reduce his uncertainty about the user's true location. However, the user is only aware of the adversary's knowledge and does not make any assumption about his inference attack. Hence, she prepares the protection mechanism against the most optimal attack. By modeling the problem as a Bayesian Stackelberg competition, we ensure that the optimal LPPM is designed anticipating such a strong inference attack. We have shown that our approach is superior to other LPPMs such as basic location obfuscation. The superiority of the optimal LPPM over alternatives is more significant when the service-quality constraint imposed by the user is tightened. Hence, our solution is effective exactly where it will be used. Finally, our results confirm that loosening the service-quality constraint allows for increased privacy protection, but the magnitude of this increase strongly depends on the user profile, i.e., on the degree to which a user's location is predictable from her LBS access profile.

To the best of our knowledge, this is the first framework that explicitly includes the adversarial knowledge into a privacy-preserving design process and that considers the *common knowledge* between the privacy protector and the attacker. Our solution is a promising step forward in the quest for robust and effective privacy preserving systems.

**Related Publication** [STT+12]

# Conclusion

*The plants say, "We are green of ourselves, we are gay, smiling, and blooming and we are tall (by nature)."*
*The season of summer says (to them), "O peoples, behold yourselves when I depart!"*

Rumi

In this thesis, we focus on how to quantify and protect the location privacy of mobile users, in the context of location-based services (LBSs). What users share with service providers in various LBSs can lead to the disclosure of their personal information. Therefore, privacy preserving mechanisms are used to reduce the amount of information that users reveal to third parties (e.g., service providers). We provide an analytical framework to evaluate users' location privacy. Our main emphasis is on the adversary model. We formalize the problem of quantifying location privacy as an estimation problem: (i) the adversary has some prior knowledge about users, (ii) he observes their access to an LBS that contains their (perhaps anonymized and obfuscated) location information, and (iii) he tries to estimate users' private information that is hidden from him. We build an inference engine, using hidden Markov models, to find users' identities and estimate their location traces. We implement this engine in an open-source software tool called location-privacy and mobility meter (LPM). The results of using LPM on real location traces indicate that users' location privacy not only depend on the protection mechanism, but also on the adversary's knowledge and his objectives and inference attacks. We then formalize the problem of designing location-privacy preserving mechanisms as a Stackelberg game between user and adversary. As the adversary does not know a user's actual location, yet he has a probabilistic belief about it, the game is also Bayesian. By solving the game, we find the optimal strategy for the user against an adversary who tries to minimize his estimation error (which is equivalent to the user's privacy). Our results indicate that the user's strategy, which anticipates the adversary, provides a higher level of privacy for users compared with basic obfuscation mechanisms.

This thesis is a step towards a better understanding and analysis of threats against users' location privacy. It also provides solutions for protecting their location privacy in the context of location-based services. Some research directions to continue this work are as follows.

In Chapter 1, we provide a three-layer model for locations. According to this simple model, a location has geographical coordinates; it is a type of place (e.g., a university building); and it is related to specific activities (e.g., education) that determine the type and semantic of the location. Our main focus is on the bottom (geographical) level. As future work, the users' profiles can be extended to the top (semantic) level. Each user can be associated with her interest in visiting different types of locations. Furthermore, a user's profile can model how a user moves between locations with different semantics, or how frequently and for how long she

visits different types of locations (e.g., a restaurant, a cinema). Subsequently, the protection mechanisms and inference attacks need to be extended in a corresponding manner.

In Chapter 2, we provide an algorithm for constructing a user's profile (which is the adversary's knowledge about the user). The main focus is on mobility profiles, i.e., the probability distribution over the locations that a user visits. In this thesis, we assume that users' mobilities are independent of each other. As future work, this assumption can be relaxed. Because of their personal and social ties, people spend a certain amount of their time with some specific people, e.g., their family. Therefore, an adversary can model this in the users' profiles, and take this into account in the inference attacks.

In Chapter 3, we design inference attacks against location-privacy preserving mechanisms, in order to evaluate the users' location privacy. Within the evaluation time window, we assume that users have persistent pseudonyms. Furthermore, we ignore other traces that are observed from users (perhaps with different pseudonyms) before and after the evaluation time window. As future work, we can extend our inference framework to also evaluate the effectiveness of pseudonym-change protection mechanisms. Additionally, it is worth looking into the problem of designing optimal pseudonym-change strategies for mobile users.

In Chapter 4, we focus on finding optimal defense strategies for users against localization attacks. Whenever a user accesses an LBS with an obfuscated location, her concern is to hide her current location from the adversary. Furthermore, we consider users who access the LBS sporadically. So, given a user's profile, we assume conditional independence between two subsequent locations from which she accesses the LBS. As future work, we can relax this independence assumption. Furthermore, we can apply our game theoretic methodology to find the optimal defense strategy for users against tracking attacks.

# Bibliography


[AA01] Dakshi Agrawal and Charu C. Aggarwal. On the design and quantification of privacy preserving data mining algorithms. In *Proceedings of the twentieth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, PODS '01, pages 247–255, New York, NY, USA, 2001. ACM.

[AS00] Rakesh Agrawal and Ramakrishnan Srikant. Privacy-preserving data mining. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, SIGMOD '00, pages 439–450, New York, NY, USA, 2000. ACM.

[BHV07] Levente Buttyan, Tamas Holczer, and Istvan Vajda. On the effectiveness of changing pseudonyms to provide location privacy in vanets. In *ESAS*, pages 129–141, 2007.

[BJB+12] L. Bindschaedler, M. Jadliwala, I. Bilogrevic, I. Aad, P. Ginzboorg, V. Niemi, and J.P. Hubaux. Track me if you can: On the effectiveness of context-based identifier changes in deployed mobile networks. *Proceedings of the 19th Annual Network and Distributed System Security Symposium*, 2012.

[BLPW08] Bhuvan Bamba, Ling Liu, Peter Pesti, and Ting Wang. Supporting anonymous location queries in mobile environments with privacygrid. In *WWW '08: Proceeding of the 17th international conference on World Wide Web*, pages 237–246, New York, NY, USA, 2008. ACM.

[Bra00] S.A. Brands. *Rethinking public key infrastructures and digital certificates: building in privacy.* MIT Press, 2000.

[BS03] Alastair R. Beresford and Frank Stajano. Location privacy in pervasive computing. *IEEE Pervasive Computing*, 2(1):46–55, 2003.

[BS04] Alastair R. Beresford and Frank Stajano. Mix zones: User privacy in location-aware services. In *PERCOMW '04: Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications Workshops*, page 127, Washington, DC, USA, 2004. IEEE Computer Society.

[BS11] Michael Brückner and Tobias Scheffer. Stackelberg games for adversarial prediction problems. In Chid Apté, Joydeep Ghosh, and Padhraic Smyth, editors, *17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2011)*, pages 547–555. ACM, 2011.

[BSVD09] S. Buchegger, D. Schiöberg, L.H. Vu, and A. Datta. Peerson: P2p social networking: early experiences and insights. In *Proceedings of the Second ACM EuroSys Workshop on Social Network Systems*, pages 46–52. ACM, 2009.

[BWJ05] Claudio Bettini, X. Sean Wang, and Sushil Jajodia. Protecting privacy against location-based personal identification. In *In 2nd VLDB Workshop SDM*, pages 185–199, 2005.

[Can02] J. Canny. Collaborative filtering with privacy. In *Security and Privacy, 2002. Proceedings. 2002 IEEE Symposium on*, pages 45–57. IEEE, 2002.

[CG09] Richard Chow and Philippe Golle. Faking contextual data for fun, profit, and privacy. In *WPES '09: Proceedings of the 8th ACM workshop on Privacy in the electronic society*, pages 105–108, New York, NY, USA, 2009. ACM.

[CGKS95] B. Chor, O. Goldreich, E. Kushilevitz, and M. Sudan. Private information retrieval. In *Foundations of Computer Science, 1995. Proceedings., 36th Annual Symposium on*, pages 41–50. IEEE, 1995.

[Cha81] David Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM*, 24(2), February 1981.

[CKGS98] Benny Chor, Eyal Kushilevitz, Oded Goldreich, and Madhu Sudan. Private information retrieval. *J. ACM*, 45(6):965–981, November 1998.

[CML06] Chi-Yin Chow, Mohamed F. Mokbel, and Xuan Liu. A peer-to-peer spatial cloaking algorithm for anonymous location-based service. In *GIS '06: Proceedings of the 14th annual ACM international symposium on Advances in geographic information systems*, pages 171–178, New York, NY, USA, 2006. ACM.

[CMS09] L.A. Cutillo, R. Molva, and T. Strufe. Privacy preserving social networking through decentralization. In *Wireless On-Demand Network Systems and Services, 2009. WONS 2009. Sixth International Conference on*, pages 145–152. IEEE, 2009.

[CPHL07] Giorgio Calandriello, Panos Papadimitratos, Jean-Pierre Hubaux, and Antonio Lioy. Efficient and robust pseudonymous authentication in vanet. In *VANET '07: Proceedings of the fourth ACM international workshop on Vehicular ad hoc networks*, pages 19–28, New York, NY, USA, 2007. ACM.

[CPP08] K. Chatzikokolakis, C. Palamidessi, and P. Panangaden. Anonymity protocols as noisy channels. *Information and Computation*, 206(2):378–401, 2008.

[CT06] Thomas M. Cover and Joy A. Thomas. *Elements of information theory, 2nd Edition*. Wiley-Interscience, 2006.

[CVH02] J. Camenisch and E. Van Herreweghen. Design and implementation of the idemix anonymous credential system. In *Proceedings of the 9th ACM conference on Computer and communications security*, pages 21–30. ACM, 2002.

[Dan04] George Danezis. Designing and attacking anonymous communication systems. Technical Report UCAM-CL-TR-594, University of Cambridge, Computer Laboratory, July 2004.

[DDM03] George Danezis, Roger Dingledine, and Nick Mathewson. Mixminion: Design of a Type III Anonymous Remailer Protocol. In *Proceedings of the 2003 IEEE Symposium on Security and Privacy*, pages 2–15, May 2003.

[DMDBP08] Yoni De Mulder, George Danezis, Lejla Batina, and Bart Preneel. Identification via location-profiling in gsm networks. In *WPES '08: Proceedings of the 7th ACM workshop on Privacy in the electronic society*, pages 23–32, New York, NY, USA, 2008. ACM.

[DMNS06] C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. *Theory of Cryptography*, pages 265–284, 2006.

[DMS04] Roger Dingledine, Nick Mathewson, and Paul Syverson. Tor: the second-generation onion router. In *Proceedings of the 13th conference on USENIX Security Symposium - Volume 13*, SSYM'04, pages 21–21, Berkeley, CA, USA, 2004. USENIX Association.

[DPV08] Sanjoy Dasgupta, Christos Papadimitriou, and Umesh Vazirani. *Algorithms*. McGraw-Hill, New York, NY, 2008.

[DR92] J. Diebolt and C. P. Robert. Estimation of finite mixture distributions through bayesian sampling. *Journal of the Royal Statistical Society: Series B*, 1992.

[DSCP02] Claudia Diaz, Stefaan Seys, Joris Claessens, and Bart Preneel. Towards measuring anonymity. In Roger Dingledine and Paul Syverson, editors, *Proceedings of Privacy Enhancing Technologies Workshop (PET 2002)*. Springer-Verlag, LNCS 2482, April 2002.

[Dwo06] C. Dwork. Differential privacy. *Automata, languages and programming*, pages 1–12, 2006.

[Dwo08] C. Dwork. Differential privacy: A survey of results. *Theory and Applications of Models of Computation*, pages 1–19, 2008.

[DZ03] W. Du and Z. Zhan. Using randomized response techniques for privacy-preserving data mining. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 505–510. ACM, 2003.

[FDW12] Drew Fisher, Leah Dorner, and David Wagner. Short paper: Location privacy: User behavior in the field. In *2nd Annual ACM CCS Workshop on Security and Privacy in Smartphones and Mobile Devices (SPSM)*, 2012.

[FM02] Michael J. Freedman and Robert Morris. Tarzan: A peer-to-peer anonymizing network layer. In *Proceedings of the 9th ACM Conference on Computer and Communications Security (CCS 2002)*, Washington, DC, November 2002.

[FMHP09] Julien Freudiger, Mohammad Hossein Manshaei, Jean-Pierre Hubaux, and David C. Parkes. On non-cooperative location privacy: a game-theoretic analysis. In *CCS '09: Proceedings of the 16th ACM conference on Computer and communications security*, pages 324–337, New York, NY, USA, 2009. ACM.

[FPIU10] Olumofin Femi, K. Tysowski Piotr, Goldberg Ian, and Hengartner Urs. Achieving efficient query privacy for location based services. In *Privacy Enhancement Technologies (PETS)*, 2010.

[FSH09] Julien Freudiger, Reza Shokri, and Jean-Pierre Hubaux. On the optimal placement of mix zones. In *PETS '09: Proceedings of the 9th International Symposium on Privacy Enhancing Technologies*, pages 216–234, Berlin, Heidelberg, 2009. Springer-Verlag.

[FSH11] Julien Freudiger, Reza Shokri, and Jean-Pierre Hubaux. Evaluating the privacy risk of location-based services. In *Financial Cryptography and Data Security (FC)*, 2011.

[GDV08] Aris Gkoulalas-Divanis and Vassilios S. Verykios. A free terrain model for trajectory k—anonymity. In *DEXA '08: Proceedings of the 19th international conference on Database and Expert Systems Applications*, pages 49–56, Berlin, Heidelberg, 2008. Springer-Verlag.

[Gey91] C. Geyer. Markov chain Monte-Carlo maximum likelihood. In *Computing Science and Statistics, Proceedings of the 23rd Symposium on the Interface*, 1991.

[GG84] S. Geman and D. Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (6):721–741, 1984.

[GG03] Marco Gruteser and Dirk Grunwald. Anonymous usage of location-based services through spatial and temporal cloaking. In *MobiSys '03: Proceedings of the 1st international conference on Mobile systems, applications and services*, pages 31–42, New York, NY, USA, 2003. ACM.

[GL05] Bugra Gedik and Ling Liu. Location privacy in mobile systems: A personalized anonymization model. In *ICDCS '05: Proceedings of the 25th IEEE International Conference on Distributed Computing Systems*, pages 620–629, Washington, DC, USA, 2005. IEEE Computer Society.

[GL08] Buğra Gedik and Ling Liu. Protecting location privacy with personalized k-anonymity: Architecture and algorithms. *IEEE Transactions on Mobile Computing*, 7(1):1–18, 2008.

[GP09] Philippe Golle and Kurt Partridge. On the anonymity of home/work location pairs. In *Pervasive '09: Proceedings of the 7th International Conference on Pervasive Computing*, pages 390–397, Berlin, Heidelberg, 2009. Springer-Verlag.

[GRS99] D. Goldschlag, M. Reed, and P. Syverson. Onion routing. *Communications of the ACM*, 42(2):39–41, 1999.

[Gut02] Serge Gutwirth. *Privacy and the Information Age*. Rowman and Littlefield Publishers, 2002.

[Has70] W.K. Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

[HGH+08] Baik Hoh, Marco Gruteser, Ryan Herring, Jeff Ban, Daniel Work, Juan-Carlos Herrera, Alexandre M. Bayen, Murali Annavaram, and Quinn Jacobson. Virtual trip lines for distributed privacy-preserving traffic monitoring. In *MobiSys '08: Proceeding of the 6th international conference on Mobile systems, applications, and services*, pages 15–28, New York, NY, USA, 2008. ACM.

[HGXA06] Baik Hoh, Marco Gruteser, Hui Xiong, and Ansaf Alrabady. Enhancing security and privacy in traffic-monitoring systems. *IEEE Pervasive Computing*, 5(4):38–46, 2006.

[HGXA07] Baik Hoh, Marco Gruteser, Hui Xiong, and Ansaf Alrabady. Preserving privacy in gps traces via uncertainty-aware path cloaking. In *CCS '07: Proceedings of the 14th ACM conference on Computer and communications security*, pages 161–171, New York, NY, USA, 2007. ACM.

[HVCT10] Nicholas Hopper, Eugene Y. Vasserman, and Eric Chan-Tin. How much anonymity does network latency leak? *ACM Transactions on Information and System Security*, 13(2), 2010.

[HYMS05] Leping Huang, Hiroshi Yamane, Kanta Matsuura, and Kaoru Sezaki. Towards modeling wireless location privacy. In *Proceedings of Privacy Enhancing Technologies Workshop (PET 2005)*, pages 59–77, 2005.

[HYMS06] Leping Huang, Hiroshi Yamane, Kanta Matsuura, and Kaoru Sezaki. Silent cascade: Enhancing location privacy without communication qos degradation. In *Security of Pervasive Computing (SPC)*, pages 165–180, 2006.

[JWH07] Tao Jiang, Helen J. Wang, and Yih-Chun Hu. Preserving location privacy in wireless lans. In *MobiSys '07: Proceedings of the 5th international conference on Mobile systems, applications and services*, pages 246–257, New York, NY, USA, 2007. ACM.

[KGMP07] P. Kalnis, G. Ghinita, K. Mouratidis, and D. Papadias. Preventing location-based identity inference in anonymous spatial queries. *Knowledge and Data Engineering, IEEE Transactions on*, 19(12):1719–1733, Dec. 2007.

[Kru07] John Krumm. Inference attacks on location tracks. In *In Proceedings of the Fifth International Conference on Pervasive Computing (Pervasive), volume 4480 of LNCS*, pages 127–143. Springer-Verlag, 2007.

[Kru09a] John Krumm. Realistic driving trips for location privacy. In *Pervasive '09: Proceedings of the 7th International Conference on Pervasive Computing*, pages 25–41, Berlin, Heidelberg, 2009. Springer-Verlag.

[Kru09b] John Krumm. A survey of computational location privacy. *Personal Ubiquitous Comput.*, 13(6):391–399, 2009.

[KT75] S. Karlin and H.E. Taylor. *A first course in stochastic processes*. Academic press, 1975.

[KYK+11] D. Korzhyk, Z. Yin, C. Kiekintveld, V. Conitzer, and M. Tambe. Stackelberg vs. Nash in security games: An extended investigation of interchangeability, equivalence, and uniqueness. *Journal of Artificial Intelligence Research*, 41:297–327, May–August 2011.

[KYS05] H. Kido, Y. Yanagisawa, and T. Satoh. An anonymous communication technique using dummies for location-based services. In *Pervasive Services, 2005. ICPS '05. Proceedings. International Conference on*, pages 88–97, July 2005.

[LB10] Jean-Yves Le Boudec. *Performance Evaluation of Computer and Communication Systems*. EPFL Press, Lausanne, Switzerland, 2010.

[LC09] Wei Liu and Sanjay Chawla. A game theoretical model for adversarial learning. In Yücel Saygin, Jeffrey Xu Yu, Hillol Kargupta, Wei Wang, Sanjay Ranka, Philip S. Yu, and Xindong Wu, editors, *IEEE International Conference on Data Mining Workshops (ICDM 2009)*, pages 25–30. IEEE Computer Society, 2009.

[LJY08] Hua Lu, Christian S. Jensen, and Man Lung Yiu. Pad: privacy-area aware, dummy-based location privacy in mobile services. In *MobiDE '08: Proceedings of the Seventh ACM International Workshop on Data Engineering for Wireless and Mobile Access*, pages 16–23, New York, NY, USA, 2008. ACM.

[LP00] Y. Lindell and B. Pinkas. Privacy preserving data mining. In *Advances in Cryptology—CRYPTO 2000*, pages 36–54. Springer, 2000.

[LPM12] Location-Privacy and Mobility Meter tool. Available online through http://icapeople.epfl.ch/rshokri, 2012.

[LSHP06] Mingyan Li, Krishna Sampigethaya, Leping Huang, and Radha Poovendran. Swing & swap: user-centric approaches towards maximizing location privacy. In *WPES '06: Proceedings of the 5th ACM workshop on Privacy in electronic society*, pages 19–28, New York, NY, USA, 2006. ACM.

[Mac03] David J. C. MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, 2003.

[MCA06] Mohamed F. Mokbel, Chi-Yin Chow, and Walid G. Aref. The new casper: query processing for location services without compromising privacy. In *VLDB '06: Proceedings of the 32nd international conference on Very large data bases*, pages 763–774. VLDB Endowment, 2006.

[MD05] Steven J. Murdoch and George Danezis. Low-cost traffic analysis of Tor. In *Proceedings of the 2005 IEEE Symposium on Security and Privacy*, May 2005.

[MPS13] Prateek Mittal, Charalampos Papamanthou, and Dawn Song. Preserving link privacy in social network based systems. In *NDSS*, 2013.

[MRC09] Joseph Meyerowitz and Romit Roy Choudhury. Hiding stars with fireworks: location privacy through camouflage. In *MobiCom '09: Proceedings of the 15th annual international conference on Mobile computing and networking*, pages 345–356, New York, NY, USA, 2009. ACM.

[MT07] F. McSherry and K. Talwar. Mechanism design via differential privacy. In *Foundations of Computer Science, 2007. FOCS'07. 48th Annual IEEE Symposium on*, pages 94–103. IEEE, 2007.

[MYYR10] Chris Y.T. Ma, David K.Y. Yau, Nung Kwan Yip, and Nageswara S.V. Rao. Privacy vulnerability of published anonymous mobility traces. In *Proceedings of the sixteenth annual international conference on Mobile computing and networking*, MobiCom '10, pages 185–196, New York, NY, USA, 2010. ACM.

[MZA+11] M.H. Manshaei, Q. Zhu, T. Alpcan, T. Basar, and J.-P. Hubaux. Game theory meets network security and privacy. *ACM Computing Surveys*, 2011.

[NAS08] Mehmet Ercan Nergiz, Maurizio Atzori, and Yucel Saygin. Towards trajectory anonymization: a generalization-based approach. In *SPRINGL '08: Proceedings of the SIGSPATIAL ACM GIS 2008 International Workshop on Security and Privacy in GIS and LBS*, pages 52–61, New York, NY, USA, 2008. ACM.

[NS08] Arvind Narayanan and Vitaly Shmatikov. Robust de-anonymization of large sparse datasets. In *Proceedings of the 2008 IEEE Symposium on Security and Privacy*, SP '08, pages 111–125, Washington, DC, USA, 2008. IEEE Computer Society.

[NS09] Arvind Narayanan and Vitaly Shmatikov. De-anonymizing social networks. In *SP '09: Proceedings of the 2009 30th IEEE Symposium on Security and Privacy*, pages 173–187, Washington, DC, USA, 2009. IEEE Computer Society.

[NS10] Arvind Narayanan and Vitaly Shmatikov. Myths and fallacies of "personally identifiable information". *Commun. ACM*, 53(6):24–26, June 2010.

[NTL+11] A. Narayanan, N. Thiagarajan, M. Lakhani, M. Hamburg, and D. Boneh. Location privacy via private proximity testing. In *Proc. of NDSS*, volume 2011, 2011.

[PBP10] S. Papadopoulos, S. Bakiras, and D. Papadias. Nearest neighbor search with strong location privacy. *Proceedings of the VLDB Endowment*, 3(1-2):619–629, 2010.

[PC02] D. Phillips and M. Curry. Privacy and the phenetic urge: Geodemographics and the changing spatiality of local practice. *Surveillance as Social Sorting: Privacy*, 2002.

[PK08] Andreas Pfitzmann and Marit Köhntopp. Anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management - a consolidated proposal for terminology., 2008.

[PPM+08] Praveen Paruchuri, Jonathan P. Pearce, Janusz Marecki, Milind Tambe, Fernando Ordóñez, and Sarit Kraus. Efficient algorithms to solve Bayesian Stackelberg games for security applications. In Dieter Fox and Carla P. Gomes, editors, *23rd AAAI Conference on Artificial Intelligence (AAAI 2008)*, pages 1559–1562. AAAI Press, 2008.

[PSDG] Michal Piorkowski, Natasa Sarafijanovic-Djukic, and Matthias Grossglauser. CRAWDAD data set epfl/mobility (v. 2009-02-24). Downloaded from http://crawdad.cs.dartmouth.edu/epfl/mobility.

[Rab89] Lawrence R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[RCD93] Christian P. Robert, Gilles Celeux, and Jean Diebolt. Bayesian estimation of hidden Markov chains: A stochastic implementation. *Statistics & Probability Letters*, 16(1):77–83, 1993.

[RR98] Michael K. Reiter and Aviel D. Rubin. Crowds: anonymity for web transactions. *ACM Trans. Inf. Syst. Secur.*, 1(1):66–92, November 1998.

[RSH10] Maxim Raya, Reza Shokri, and Jean-Pierre Hubaux. On the tradeoff between trust and privacy in wireless ad hoc networks. In *Proceedings of the third ACM conference on Wireless network security*, WiSec '10, pages 75–80, New York, NY, USA, 2010. ACM.

[SD02] Andrei Serjantov and George Danezis. Towards an information theoretic metric for anonymity. In Roger Dingledine and Paul Syverson, editors, *Proceedings of Privacy Enhancing Technologies Workshop (PET 2002)*. Springer-Verlag, LNCS 2482, April 2002.

[SFH10] Reza Shokri, Julien Freudiger, and Jean-Pierre Hubaux. A unified framework for location privacy. In *3rd Hot Topics in Privacy Enhancing Technologies (HotPETs)*, 2010.

[SFJH09] Reza Shokri, Julien Freudiger, Murtuza Jadliwala, and Jean-Pierre Hubaux. A distortion-based metric for location privacy. In *WPES '09: Proceedings of the 8th ACM workshop on Privacy in the electronic society*, pages 21–30, New York, NY, USA, 2009. ACM.

[Sha48] Claude E. Shannon. A mathematical theory of communication. *The Bell system technical journal*, 1948.

[SHL+05] Krishna Sampigethaya, Leping Huang, Mingyan Li, Radha Poovendran, Kanta Matsuura, , and Kaoru Sezaki. Caravan: Providing location privacy for vanet. In *The 3rd workshop on Embedded Security in Cars (ESCAR)*, 2005.

[SHSH11] Francisco Santos, Mathias Humbert, Reza Shokri, and Jean-Pierre Hubaux. Collaborative Location Privacy with Rational Users. In *2nd Conference on Decision and Game Theory for Security (GameSec)*, 2011. The original publication is available at www.springerlink.com.

[SPTH09] Reza Shokri, Pedram Pedarsani, George Theodorakopoulos, and Jean-Pierre Hubaux. Preserving privacy in collaborative filtering through distributed aggregation of offline profiles. In *Proceedings of the third ACM conference on Recommender systems*, RecSys '09, pages 157–164, New York, NY, USA, 2009. ACM.

[SPTH11] Reza Shokri, Panagiotis Papadimitratos, Georgios Theodorakopoulos, and Jean-Pierre Hubaux. Collaborative Location Privacy. In *8th IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS)*, 2011.

[STD+10] Reza Shokri, Carmela Troncoso, Claudia Diaz, Julien Freudiger, and Jean-Pierre Hubaux. Unraveling an old cloak: k-anonymity for location privacy. In *Proceedings of the 9th annual ACM workshop on Privacy in the electronic society*, WPES '10, pages 115–118, New York, NY, USA, 2010. ACM.

[STD+11] Reza Shokri, George Theodorakopoulos, George Danezis, Jean-Pierre Hubaux, and Jean-Yves Le Boudec. Quantifying location privacy: the case of sporadic location exposure. In *Proceedings of the 11th international conference on Privacy enhancing technologies*, PETS'11, pages 57–76, Berlin, Heidelberg, 2011. Springer-Verlag.

[STLBH11] Reza Shokri, Georgios Theodorakopoulos, Jean-Yves Le Boudec, and Jean-Pierre Hubaux. Quantifying Location Privacy. In *2011 Ieee Symposium On Security And Privacy (Sp 2011)*, IEEE Symposium on Security and Privacy, pages 247–262. Ieee Computer Soc Press, Customer Service Center, Po Box 3014, 10662 Los Vaqueros Circle, Los Alamitos, Ca 90720-1264 Usa, 2011.

[STT+12] Reza Shokri, George Theodorakopoulos, Carmela Troncoso, Jean-Pierre Hubaux, and Jean-Yves Le Boudec. Protecting location privacy: Optimal strategy against localization attacks. In *19th ACM Conference on Computer and Communications Security*. ACM, 2012.

[Swe02] L. Sweeney. k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(05):557–570, 2002.

[TD09] Carmela Troncoso and George Danezis. The bayesian traffic analysis of mix networks. In *Proceedings of the 16th ACM conference on Computer and communications security*, CCS '09, pages 369–379, New York, NY, USA, 2009. ACM.

[TGPV08] Carmela Troncoso, Benedikt Gierlichs, Bart Preneel, and Ingrid Verbauwhede. Perfect matching disclosure attacks. In *Proceedings of the 8th international symposium on Privacy Enhancing Technologies*, PETS '08, pages 2–23, Berlin, Heidelberg, 2008. Springer-Verlag.

[Tie91] L. Tierney. Markov chains for exploring posterior distributions. In *Computing Science and Statistics, Proceedings of the 23rd Symposium on the Interface*, 1991.

[Tro11] Carmela Troncoso. *Design and analysis methods for privacy technologies*. PhD thesis, Katholieke Universiteit Leuven, 2011.

[WCM09] Charles V. Wright, Scott E. Coull, and Fabian Monrose. Traffic Morphing: An Efficient Defense Against Statistical Traffic Analysis. In *NDSS*. The Internet Society, 2009.

[XC08] T. Xu and Ying Cai. Exploring historical location data for anonymity preservation in location-based services. In *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, pages 547–555, April 2008.

[Yac] Yacy - the peer to peer search engine. http://yacy.net.

[YPL07] Tun-Hao You, Wen-Chih Peng, and Wang-Chien Lee. Protecting moving trajectories with dummies. In *Mobile Data Management, 2007 International Conference on*, pages 278–282, May 2007.

[ZB11] Hui Zang and Jean Bolot. Anonymization of location data does not work: a large-scale measurement study. In *Proceedings of the 17th annual international conference on Mobile computing and networking*, MobiCom '11, pages 145–156, New York, NY, USA, 2011. ACM.

[ZGH07] G. Zhong, I. Goldberg, and U. Hengartner. Louis, lester and pierre: Three protocols for location privacy. In *Privacy Enhancing Technologies*, pages 62–76. Springer, 2007.

# Index

# Reza Shokri

**address**: EPFL IC LCA1, BC 206, Station 14, Lausanne 1015, Switzerland
**e-mail**: reza.shokri@epfl.ch
**web**: `http://icapeople.epfl.ch/rshokri`

## RESEARCH INTERESTS

Location Privacy; Computer Security and Privacy; Mobile Networks; Hidden Markov Models; Bayesian Inference; Probabilistic Modeling; Algorithms and Machine Learning

## EDUCATION

2007 – 2012 **PhD** Computer and Communication Science, EPFL, Switzerland
*Quantifying and Protecting Location Privacy*, Prof. Jean-Pierre Hubaux

2004 – 2007 **MSc** Software Computer Engineering, University of Tehran, Iran
*Anonymous Routing in Mobile Ad Hoc Networks*, Prof. Naser Yazdani

1998 – 2003 **BSc** Software Computer Engineering, University of Isfahan, Iran
*Scanning Security Vulnerabilities*, Prof. Naser Movahedi Nia, Prof. Behrouz Tork Ladani

## HONORS

Runner-up for the Award for Outstanding Research in Privacy Enhancing Technologies 2012, for "Quantifying Location Privacy" paper, in IEEE Symposium on Security and Privacy 2011.

## ACADEMIC POSITIONS

2007 – Present **Research Assistant** LCA1, EPFL, Lausanne, Switzerland

2005 – 2007 **Research Assistant** Router Laboratory, University of Tehran, Tehran, Iran

## PROFESSIONAL ACTIVITIES

Program co-chair of HotPETs 2013
Program committee member of WPES 2012, PETS 2013
Reviewer of ACM CCS, PETS, ESORICS, ACM WiSec conferences, and ACM TISSEC, IEEE TMC, IEEE TWC, IEEE TIFS, IEEE TPDS, Elsevier Comcom journals

## INVITED TALKS and SELECTED CONFERENCE PRESENTATIONS

*Protecting Location Privacy: Optimal Strategy against Localization Attacks* at CCS 2012.

*Quantifying Location Privacy: The Case of Sporadic Location Exposure* at PETS 2011.

*Quantifying Location Privacy* at IEEE S&P, Oakland, 2011.

*Quantifying Location Privacy* at Palo Alto Research Center (PARC), 2011

*Quantifying Location Privacy* at Computer Science Department, UIUC, 2011

*Location Privacy: Threats and Countermeasures* at COSIC, K.U.Leuven, 2010

*A Practical Secure Neighbor Verification Protocol for WSNs* at WiSec 2009

*Preserving Privacy in Collaborative Filtering ...* at RecSys 2009

*Location-Privacy Metrics* at WINLAB, ECE Department, Rutgers University, 2009

*Anonymous Routing in Mobile Networks* at INRIA, LIX, Ecole Polytechnique France, 2006

## TEACHING (ASSISTANCE)

*Mobile Networks*, and *Security and Cooperation in Wireless Networks*, EPFL. 2009-12

(Lecturer) *Operating System Lab.: Linux kernel programming*, Univ. of Tehran. 2005-06

*Computer Networks*, *Operating Systems*, and *Compilers*, Univ. of Isfahan. 2001

## SUPERVISED STUDENTS

Pierre Pfister, **Impact of Human Mobility on Location Privacy**. Fall 2012

Vincent Bindschaedler, **Impact of Human Mobility on Location Privacy**. Spring 2012

Saeid Sahraei, **Lower Bounds on Location Privacy**. Spring 2012

Ypatia Tsavliri, and Vasileios Agrafiotis, **Location-tagged Info. in Facebook**. Fall 2011

David Freiburghaus, **Lower Bounds on Location Privacy**. Fall 2011

Quentin Alban Hounkpatin, **Evaluating Location Obfuscation Mechanisms**. Fall 2011

Arun Mallya, **Reconstructing Noisy Trajectories**. Spring 2011

Vincent Bindschaedler, **Developing the Location-Privacy Meter Tool**. Spring 2011

Ehsan Kazemi, **Quantifying Location-Privacy**. Spring 2011

Francisco Santos, **Game Theoretic Analysis of MobiCrowd**. Spring 2011

Jean Biollay, **Privacy-Preserving Mobile Recommender Systems**. Fall 2010

Selma Chouaki, **Privacy vs. Trust in Participatory Sensing Systems**. Fall 2010

Acacio Martins, and Emanuel Cino, **A Privacy-Preserving Friend-Finder**. Spring 2010

Hai Ly Hoang, **Evaluating Location-Privacy Preserving Mechanisms**. Fall 2009

Loic Pfister, **Implementing A Collaborative Privacy Protection Method**. Fall 2009

Frederico Venturieri, **Wireless Peer-to-Peer Communication Helps Privacy**. Fall 2009

Antoine Parisod, **Privacy Preserving Recommender Systems**. Spring 2009

Laurent Bindschaedler, and Marc Bailly, **Secure Communication over SMS**. Spring 2009

Hai Ly Hoang, **Group Trust in Mobile Networks**. Spring 2009

Nawfal Cherqui, **Secure Communication in Ad-hoc Networks**. Spring 2009

Seyyd Hasan Mirjalili, **Privacy-Preserving People-centric Sensing**. Fall 2008

Loic Pfister, **Wormhole Attack Prevention in Sensor Networks**. Fall 2008

Gael Ravot, **Secure Neighbor-Verification in Sensor Networks**. Spring 2008

## SOFTWARE TOOL

*Location Privacy and Mobility Meter (LPM)* is an open-source tool developed in C++ to quantify mobility and privacy of mobile users. It models mobility of people as a time-dependent Markov chain. It provides a high-level API to learn mobility model of individuals from their (potentially incomplete) actual traces, to evaluate the randomness and similarity of their mobilities, and to quantify their location privacy under user-specified location-bases services and location-privacy preserving mechanisms (such as anonymization and obfuscation). LPM is mainly based on a Bayesian inference framework, and makes use of a variety of algorithms and statistical methods. For profiling mobile users and learning their mobility models we make use of Gibbs sampling that allows us to learn from incomplete location traces. To quantify location privacy of mobile users, we run inference attacks (such as de-anonymization, localization, and tracking) on pseudonymous/obfuscated location traces. We then compute location privacy as the expected inference error (by comparing the reconstructed traces with the actual traces). To implement the inference attacks, we make use of various iterative algorithms and dynamic programming methods of hidden Markov models (HMMs). The software plus documentations are available online through: `http://icapeople.epfl.ch/rshokri/lpm`.

## MAIN PEER-REVIEWED PUBLICATIONS

Reza Shokri, George Theodorakopoulos, Carmela Troncoso, Jean-Pierre Hubaux, and Jean-Yves Le Boudec. **Protecting Location Privacy: Optimal Strategy against Localization Attacks**. *In The 19th ACM Conference on Computer and Communications Security (CCS), Raleigh, NC, USA, 2012.*

Reza Shokri, George Theodorakopoulos, Jean-Yves Le Boudec, and Jean-Pierre Hubaux. **Quantifying Location Privacy**. *In IEEE Symposium on Security and Privacy (S&P), Oakland, CA, USA, 2011.*

Reza Shokri, George Theodorakopoulos, George Danezis, Jean-Pierre Hubaux, and Jean-Yves Le Boudec. **Quantifying Location Privacy: The Case of Sporadic Location Exposure**. *In The 11th Privacy Enhancing Technologies Symposium (PETS), Waterloo, Canada, 2011.*

Reza Shokri, Panos Papadimitratos, George Theodorakopoulos, and Jean-Pierre Hubaux. **Collaborative Location Privacy**. *8th IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS), Valencia, Spain, 2011.*

Julien Freudiger, Reza Shokri, Jean-Pierre Hubaux. **On the Optimal Placement of Mix Zones**. *In Privacy Enhancement Technologies Symposium (PETS), Seattle, WA, USA, 2009.*

Reza Shokri, Marcin Poturalski, Gael Ravot, Panos Papadimitratos, and Jean-Pierre Hubaux. **A Practical Secure Neighbor Verification Protocol for Wireless Sensor Networks**. *In Second ACM Conference on Wireless Network Security (WiSec), Zurich, Switzerland, 2009.*

Reza Shokri, Pedram Pedarsani, George Theodorakopoulos, and Jean-Pierre Hubaux. **Preserving Privacy in Collaborative Filtering through Distributed Aggregation of Offline Profiles**. *In The 3rd ACM Conference on Recommender Systems (RecSys), New York, NY, USA, 2009.*

Reza Shokri, Julien Freudiger, Murtuza Jadliwala, and Jean-Pierre Hubaux. **A Distortion-based Metric for Location Privacy**. *In ACM Workshop on Privacy in the Electronic Society (WPES), Chicago, IL, USA, 2009.*

## OTHER PEER-REVIEWED PUBLICATIONS

Francisco Santos, Mathias Humbert, Reza Shokri and Jean-Pierre Hubaux. **Collaborative Location Privacy with Rational Users**. *2nd ACM Conference on Decision and Game Theory for Security (GameSec), Maryland, USA, 2011.*

Julien Freudiger, Reza Shokri, and Jean-Pierre Hubaux. **Evaluating the Privacy Risk of Location-Based Services**. *In Financial Cryptography and Data Security (FC), St. Lucia, 2011.*

Maxim Raya, Reza Shokri, and Jean-Pierre Hubaux. **On the Tradeoff between Trust and Privacy in Wireless Ad Hoc Networks**. *In 3rd ACM Conference on Wireless Network Security (WiSec), Hoboken, NJ, USA, 2010.*

Reza Shokri, Carmela Troncoso, Claudia Diaz, Julien Freudiger, and Jean-Pierre Hubaux. **Unraveling an Old Cloak: k-anonymity for Location Privacy**. *In ACM Workshop on Privacy in the Electronic Society (WPES), Chicago, IL, USA, 2010.*

Reza Shokri, Julien Freudiger, and Jean-Pierre Hubaux. **A Unified Framework for Location Privacy**. *In 3rd Workshop on Hot Topics in Privacy Enhancing Technologies (HotPETs), Berlin, Germany, 2010.*

Foroogh Anoosha, Reza Shokri, Nasser Yazdani, and Amir Nayyeri. **ANTMIG: A Novel Code Migration Method to Conserve Energy in Wireless Sensor Networks**. *In Proceedings of The IEEE Wireless Communications and Networking Conference (WCNC), Las Vegas, NV, USA, 2008.*

Reza Shokri, Maysam Yabandeh, Nasser Yazdani, and Ahmad Khonsari. **Anonymous Data Delivery in MANETs using Pseudonymity in Chain-based Routing**. *In the CSI Journal on Computer Science and Engineering (CSI-JCSE), Iran, 2007.*

Reza Shokri, Amir Nayyeri, Nasser Yazdani, and Panos Papadimitratos. **Efficient and Adjustable Recipient Anonymity in Mobile Ad Hoc Networks**. *In Proceedings of The Fourth IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS), Pisa, Italy, 2007.*

Maysam Yabandeh, Reza Shokri, and Nasser Yazdani. **Formal Verification of Chain-based Anonymous Routing for Wireless Ad Hoc Networks**. *In Proceedings of International Symposium on Fundamentals of Software Engineering (FSEN), Tehran, Iran, 2007.*

Reza Shokri, Maysam Yabandeh, and Nasser Yazdani. **Anonymous Routing in MANETs using Random Identifiers**. *In Proceedings of The 6th IARIA International Conference on Networking (ICN), Sainte-Luce, Martinique, 2007.*

Reza Shokri, Nasser Yazdani, and Ahmad Khonsari. **Chain-based Anonymous Routing for Wireless Ad Hoc Networks**. *In Proceedings of 4th IEEE Consumer Communications and Networking Conference (CCNC), Las Vegas, NV, USA, 2007.*

Hossein Mohammadi, Mahmood Hasanlou, Nasser Yazdani, Ali Movaghar, and Reza Shokri. **Supporting High Mobility by End-Host Assistance - Preventing Overrunning of Routing Protocols**. *In Proceedings of IEEE Asia-Pacific Conference on Communications (APCC), Bangkok, Thailand, 2007.*

Reza Shokri, Ali Varshovi, Hossein Mohammadi, Nasser Yazdani, and Babak Sadeghian. **DDPM: Dynamic Deterministic Packet Marking for IP Traceback**. *In 14th IEEE International Conference on Networks (ICON), Singapore, 2006.*

Reza Shokri, Farhad Oroumchian, and Nasser Yazdani. **CluSID: a Clustering Method for Intrusion Detection, Improved by Information Theory**. *In Proceedings of 13th IEEE International Conference on Networks (ICON), Kuala Lumpur, Malaysia. 2005.*

Amir Nayyeri, Reza Shokri, and Nasser Yazdani. **GAAM: An Energy Conservation Method Using Code Migration for Ad hoc Sensor Networks**. *In Proceedings of 13th IEEE International Conference on Networks (ICON), Kuala Lumpur, Malaysia, 2005.*

Reza Shokri, Amir Nayyeri, and Nasser Yazdani. **Using Code Migration for Energy Conservation in Ad Hoc Sensor Networks**. *In Proceedings of 3rd International Symposium on Telecommunications (IST), Shiraz, Iran, 2005.*