

Securing 5G Positioning and its Services with Privacy Preservation

SHUSHU LIU

Securing 5G Positioning and its Services with Privacy Preservation

SHUSHU LIU

A doctoral thesis completed for the degree of Doctor of Science (Technology) to be defended, with the permission of the Aalto University School of Electrical Engineering, at a public examination held at the lecture hall AS 1 of the school on 21st December 2022 at 12:15 pm.

**Aalto University
School of Electrical Engineering
Communication and Networking
Network Security and Trust**

Supervising professor

Professor Raimo Kantola, Aalto University, Finland

Thesis advisor

Professor Zheng Yan, Xi'dian University, China

Preliminary examiners

Professor Simona Lohan, Tampere University, Finland

Associate Professor Sheng Wen, Swinburne University of Technology, Australia

Opponents

Professor Mridula Singh, Helmholtz Center for Information Security, Germany

Professor Simona Lohan, Tampere University, Finland

Aalto University publication series

DOCTORAL THESES 202/2022

© 2022 SHUSHU LIU

ISBN 978-952-64-1088-3 (printed)

ISBN 978-952-64-1089-0 (pdf)

ISSN 1799-4934 (printed)

ISSN 1799-4942 (pdf)

<http://urn.fi/URN:ISBN:978-952-64-1089-0>

Unigrafia Oy

Helsinki 2022

Finland



Author

SHUSHU LIU

Name of the doctoral thesis

Securing 5G Positioning and its Services with Privacy Preservation

Publisher School of Electrical Engineering**Unit** Communication and Networking**Series** Aalto University publication series DOCTORAL THESES 202/2022**Field of research** Security and Privacy Protection in 5G Positioning**Manuscript submitted** 5 December 2022**Date of the defence** 21 December 2022**Permission for public defence granted (date)** 2 December 2022**Language** English **Monograph** **Article thesis** **Essay thesis****Abstract**

Different from the global positioning system (GPS), the positioning in the fifth-generation (5G) cellular networks is measured through nearby access nodes and processed at the cloud/edge/fog devices. Owing to the availability of high-quality measurements and outsourced computation, the 5G positioning promises high precision, high reliability, wide coverage and low power consumption. The 5G positioning ecosystem relates to 5G positioning and its services. There are four main stakeholders in the ecosystem: location information service provider (LISP), location-based service provider (LBSP), user equipment (UE) with a 5G connection and the location information collaborator (LIC).

Focusing on 5G positioning and its services, the present dissertation aims to investigate and resolve the problems in the area of security, privacy and integrity. (1) The security of 5G positioning is threatened by various attacks from signal jamming and counterfeiting to malicious or untrusted devices and users. **For solving the security problem in 5G positioning, a framework composed of three modules is proposed to defend against jamming and collusion attacks.** (2) To prevent the privacy violation in outsourced 5G positioning computation, **two protocols (Pub-pos and Pri-pos) with flexible privacy selection are proposed.** (3) Also in the case of outsourced 5G positioning services, **we apply an integrity check method by creating a backdoor in an outsourced positioning model based on machine learning.** (4) LIC facilitates the position verification by interacting with nearby UEs through distributed device-to-device (D2D) communication. However, the position of private LIC is leaked in the position verification process. To solve this problem, **we propose a privacy preservation scheme implemented with the double order-preserving encryption (OPE) and a coordinate-based verification method.** Experimental results show significant performance improvement. (5) LBS provision is conducted between the UE and LBSP. For privacy protection, the UE wants to hide its position information and LBSP wants to protect its database from any unauthorized access. However, it is challenging to support a variety of LBS queries and meet the low latency requirement in LBS provision based on 5G positioning, especially when both UE position privacy and LBSP data privacy should be protected at the same time. **We propose two protocols, based on exact and fuzzy kNN queries, to achieve mutual privacy preservation, flexible keyword search and low latency.** All the proposed schemes are evaluated with simulations or real-world datasets. The results demonstrate the improvement or the trade-off among security, privacy, integrity and overhead. It is expected that this dissertation can further advance secure and privacy-preserving 5G positioning and its services.

Keywords Security, Privacy, 5G Positioning, LBS, D2D**ISBN (printed)** 978-952-64-1088-3**ISBN (pdf)** 978-952-64-1089-0**ISSN (printed)** 1799-4934**ISSN (pdf)** 1799-4942**Location of publisher** Helsinki**Location of printing** Helsinki **Year** 2022**Pages** 196**urn** <http://urn.fi/URN:ISBN:978-952-64-1089-0>

Preface

As a conclusion to this unbelievable life-changing journey as a doctoral researcher, this dissertation summarises the main outcomes I experienced from Spring 2018 to Autumn 2022 in the Department of Communications and Networking (ComNet), School of Electrical Engineering, Aalto University, Finland. The works herein would not have been possible without the invaluable help/guidance kindly offered by my supervisors, my colleagues and my family. Therefore, at this point of my doctoral dissertation, I would like to express my sincere gratitude to those incredible people in my life.

Prof. Raimo Kantola. A respectful Finnish professor with great knowledge and high level of work efficiency and who always welcomes me (and everyone) with unconditional support and guides us with effective supervision and sufficient flexibility. All of your support has laid the foundation for these high-quality works and publications. Thank You, Raimo.

Prof. Zheng Yan. An enthusiastic and energetic professor who influences me using her hard working spirit, motivating me to push myself towards an advanced level and work hard and create. Thank you for offering me this wonderful start and leading me as an independent researcher. Thank you for supporting me with abundant degrees of freedom. I have been lucky enough to learn from you, thank you.

Prof. Xueqin Liang. A kind, humble and highly capable researcher who has provided me with both technical and nontechnical help. She impressed me with her high efficiency, preciseness and sense of responsibility. My working experience with her has been such a valuable asset to me.

Dr. Wei Feng. A hungry learner who keeps exploring with curiosity and diligence and always helps with kindness. The discussions with him have always been inspiring and productive. Many thanks for your insightful and valuable comments and suggestions.

In addition, I would like to thank all the colleagues I have met along the way; your presence has made my life colourful and unpredictable. The names are not listed in any particular order: *Prof. A. Liu, Prof. W. Ding, Dr W. Wang, Dr Y. Lu, M.A. Y. Li, Dr S. Fei, Dr X. Wang, Dr Y. Xia and all the lab members.* Additionally, a special thanks goes to my precious

family members. *My parents and my sister*, thanks for your unconditional love and support. I also would like to acknowledge my sincere gratitude to *Aalto University and the Academy of Finland (TruSoNet Project)* for the financial support of my degree work. Your generous gift made has greatly helped me in pursuing my dreams. I want to thank you for helping me make a positive change in my life.

Finally, I sincerely express my gratitude to *the pre-examiners and my opponent* for your consideration and efforts and for being an indispensable part of this memorable journey of mine!

Espoo, Finland, December 6, 2022,

Shushu Liu

Contents

Preface	1
Contents	3
List of Publications	7
Author's contributions	9
List of Figures	11
List of Tables	13
Abbreviations	15
Symbols	17
1. Introduction	21
1.1 Background	22
1.1.1 5G Positioning	23
1.1.2 Verifiable Positioning	27
1.1.3 Location-based Services	28
1.2 Security and Privacy of 5G Positioning and Services . . .	29
1.2.1 Security Threats	29
1.2.2 Privacy Leakage	30
1.3 Research Problems	31
1.3.1 5G Positioning	31
1.3.2 Verifiable Positioning	33
1.3.3 Location-Based Services	34
1.4 Dissertation Work Introduction and Contributions	36
1.4.1 Security and Privacy in 5G Positioning	36
1.4.2 Verifiable Positioning	37
1.4.3 Privacy-Preserving LBS	38
1.5 Dissertation Structure	39

2.	Preliminaries and Related Work	41
2.1	Preliminaries	41
2.1.1	Order-Preserving Encryption	41
2.1.2	Paillier Cryptosystem	42
2.1.3	Garbled Circuits	42
2.1.4	Local Differential Privacy	42
2.2	Related Work	43
2.2.1	Security and Privacy in 5G Positioning	44
2.2.2	Verifiable Positioning	47
2.2.3	Privacy Protection in D2D Location Verification	48
2.2.4	Privacy-Preserving Location-Based Service	51
2.3	Summary	52
3.	Security and Privacy Protection in 5G Positioning	53
3.1	Problem Statement	53
3.2	Secure Positioning with Truth Discovery, Attack Detection and Tracing	54
3.2.1	Truth Discovery	55
3.2.2	Attack Detection	58
3.2.3	Attack Tracing	59
3.3	Efficient Privacy Protection Protocols for 5G-Enabled Po- sitioning	60
3.3.1	Pub-pos: Privacy-Preserving Positioning Based on Public Access Nodes	61
3.3.2	Pri-pos: Privacy-Preserving Positioning Based on Private Access Nodes	66
3.4	Performance Evaluation	70
3.4.1	Experimental Test of Secure Positioning Modules	70
3.4.2	Experimental Test of Privacy Protocols	75
3.5	Summary	76
4.	Verifiable Outsourced Positioning Model	77
4.1	System Model	77
4.2	Verifiable Positioning Model with the Backdoor Strategy	77
4.3	Performance Evaluation	80
4.3.1	Experimental Test of Effectiveness	81
4.3.2	Experimental Test of Trigger Data Size	81
4.3.3	Experimental Test of Trigger Data Source	82
4.4	Summary	82
5.	Privacy-Protected D2D Cooperative Location Verification	85
5.1	Problem Statement	85
5.1.1	System Model	85
5.1.2	Security Model	86
5.2	Privacy-Preserving D2D Location Verification	87

5.2.1	Coordinates-Based Location Verification	87
5.2.2	Order-Preserving Encryption-Based Location Ver- ification	88
5.3	Performance Evaluation	90
5.3.1	Experimental Setup	90
5.3.2	Experimental Results	91
5.4	Summary	92
6.	Privacy-Preserving Location-Based Service	93
6.1	Problem Statement	93
6.1.1	System Model and Assumptions	93
6.1.2	Threat Model	94
6.2	Mutual Privacy Protected LBS Query	94
6.2.1	Exact kNN Query (E-kNN) Design and Security Analysis	94
6.2.2	Fuzzy kNN Query (F-kNN) Design and Security Analysis	99
6.3	Performance Evaluation	106
6.3.1	Experimental Test of E-kNN query	106
6.3.2	Experimental Test of the F-kNN query	107
6.4	Summary	109
7.	Conclusion and Future Perspectives	111
7.1	Conclusion	111
7.2	Applicability and Limitations	112
7.3	Future Perspectives	114
	References	115
	Publications	125

List of Publications

This thesis consists of an overview and of the following publications which are referred to in the text by their Roman numerals.

- I** Shushu Liu and Zheng Yan. Efficient Privacy Protection Protocols for 5G Enabled Positioning in Industrial IoT. *IEEE Internet of Things Journal*, DOI:10.1109/JIOT.2022.3161148, Jan 2022.
- II** Shushu Liu, Zheng Yan, and Raimo Kantola. Privacy-preserving D2D Cooperative Location Verification. In *IEEE Global Communications Conference*, Madrid, 1-6, Dec 2021.
- III** Shushu Liu, and Zheng Yan. Verifiable Edge Computing for Indoor Positioning. In *IEEE International Conference on Communications, Virtual*, 1-6, June 2020.
- IV** Shushu Liu, An Liu, Zheng Yan and Wei Feng. Efficient LBS queries with mutual privacy preservation in IoV. *Vehicular Communications*, 16, 62-71, Jan 2019.
- V** Shushu Liu and Zheng Yan. Pri-CrowdLBS: Local Differential Privacy for Crowdsourcing-based LBS with Top-k Spatial-textual Query. Submitted to *ASIACCS*, Aug 2022.
- VI** Yilin Li, Shushu Liu, Zheng Yan, and Robert H. Deng. Secure 5G positioning with truth discovery, attack detection and tracing. *IEEE Internet of Things Journal*, DOI: 10.1109/JIOT.2021.3088852, June 2021.

Author's contributions

Publication I: “Efficient Privacy Protection Protocols for 5G Enabled Positioning in Industrial IoT”

The original idea, simulation code and the overall manuscript of this publication were completed by Shushu Liu. Prof. Yan gave high-level supervision. The development of the proposed idea and strategy and the results are discussed with Prof. Yan. Besides, Prof. Yan offered constructive comments on the manuscript's structure and technical ingredients.

Publication II: “Privacy-preserving D2D Cooperative Location Verification”

The original idea and objective of this publication were discussed and identified by Shushu Liu and Prof. Yan. 90% of the manuscript for this publication was contributed by Shushu Liu. Specifically, Shushu Liu produced the state of the art review, solution design and experimental evaluation. The manuscript was completed by Shushu Liu and commented with other co-authors.

Publication III: “Verifiable Edge Computing for Indoor Positioning”

The research problem and original ideas are proposed by Shushu Liu who also wrote the manuscript. The experiment is also designed and conducted by Shushu Liu. Prof. Yan contributed with insightful discussion and careful review of the whole paper.

Publication IV: “Efficient LBS queries with mutual privacy preservation in IoV”

As an extension of the previous publication, Shushu Liu identified its use case towards the IoT scenario. The solution is originally proposed by An Liu and implemented by Shushu Liu. All the authors have contributed to the technical discussion. The whole manuscript was produced by Shushu Liu and revised carefully by Prof. Yan.

Publication V: “Pri-CrowdLBS: Local Differential Privacy for Crowdsourcing-based LBS with Top-k Spatial-textual Query”

The research problem was inspired and defined by Shushu Liu. Shushu Liu also contributed 95% to scheme design, programming and experimental evaluation with the supervision from Prof. Yan. The manuscript is drafted by Shushu Liu and polished by Prof. Yan.

Publication VI: “Secure 5G positioning with truth discovery, attack detection and tracing”

As the co-first author, Shushu Liu contributes equally with first author Yilin Li. The research problem and main research methods of this paper are originally proposed by Prof. Yan and Prof. Deng. Yilin Li and Shushu Liu contributed together with the task together where Yilin Li is responsible for scheme design and simulation and Shushu Liu is responsible for manuscript writing. The solution has been discussed and examined together.

List of Figures

1.1	Illustration of 5G positioning and services	23
1.2	Illustration of a 5G UDN, where the access nodes are attached to lamp posts, and user equipment transmits periodical uplink pilot signals to the access nodes. The position estimation is carried out by a fusion center and sent back to the user equipment through downlink beams.	24
1.3	Example of trilateration-based 3D positioning	24
1.4	Example of angulation-based positioning	26
1.5	Example of RSS-pattern-match-based positioning	27
1.6	Overview of D2D cooperative location verification	27
1.7	Overview of LBS query	29
1.8	The relationship among research problems, technical background, contributions, publications and chapters	35
3.1	Secure positioning with truth discovery, attack detection and tracing	54
3.2	Analysis of the secure positioning modules in the experimental setup	71
3.3	Truth discovery under the effect of interval distance	72
3.4	Truth discovery under effect of the number of ANs	73
3.5	Computation and communication comparison of different protocols	75
4.1	Illustration of the outsourced positioning model	78
4.2	Verifiable positioning model	79
5.1	Coordinates-based location verification	87
5.2	Performance comparison under different security strengths	91
6.1	Efficiency comparison between E-kNN and Yi et al.	107
6.2	Accuracy of F-kNN with variable ϵ	108

6.3	Efficiency comparison of F-kNN with the variables m , n and q	109
7.1	The integration of solutions within 5G positioning ecosystem	12

List of Tables

1.1	Summary of the positioning mechanisms	27
2.1	Related work comparison w.r.t. 5G positioning security . .	44
2.2	Related work comparison w.r.t. 5G positioning privacy protection	46
2.3	Related work summary w.r.t verifiable positioning	47
2.4	Related work w.r.t privacy protection in D2D location verification	49
2.5	Related work summary w.r.t privacy-preserving LBS . . .	50
3.1	Notations	55
3.2	Notations	61
3.3	Parameter settings	71
3.4	Example of simulated dataset	71
3.5	Performance of attack detection with different neural networks	74
3.6	Performance of attack tracing with different neural networks	74
4.1	Parameter setting for each model	80
4.2	Statistic information of datasets	80
4.3	Effectiveness of verifiable positioning model	80
4.4	Influence of the trigger dataset size	82
4.5	Influence of trigger data source	82
5.1	Notations	86
5.2	Details of online latency in location verification	91
6.1	Notations	96
6.2	Notations	100
6.3	Summary of the dataset	106
6.4	Summary of the Yelp dataset	107
6.5	Parameters and settings	107

Abbreviations

3GPP	3rd Generation Partnership Project
5G	Fifth Generation Cellular Network
A-UE	Assistant User Equipment
AN	Access Node
APDR	Abnormal Positioning Data Ratio
APEM	AN Positioning Error Mean
APEV	AN Positioning Error Variance
APS	Abnormal Positioning Status
AUR	Abnormal Upload Ratio
CP-ABE	Ciphertext-policy Attribute based Encryption
DBSCAN	Density-based Spatial Clustering of Applications with Noise
DDH	Decision Diffie-Hellman
D2D	Device-to-Device Communication
DoA	Direction of Arrival
DP	Differential Privacy
FC	Fusion Center
GPS	Global Positioning System
IQ	Interval Query
kNN	k Nearest Neighbor
LBS	Location Based Service

L BSP	Location Based Service Provider
L DP	Local Differential Privacy
L IC	Location Information Collaborator
L ISP	Location Information Service Provider
LoS	Line of Sight
MIMO	Multiple Input and Multiple Output
ML	Machine Learning
MLP	Multilayer Perceptron
MSE	Mean Square Error
NN	Neural Network
OPE	Order Preserving Encryption
OT	Oblivious Transfer
PEM	Positioning Error Mean
PEV	Positioning Error Variance
POI	Point of Interest
QoS	Quality of Service
RF	Random Forest
RSS	Received Signal Strength
SGX	Software Guard Extensions
SMC	Secure Multi-party Computation
SVM	Support Vector Machine
TDoA	Time Difference of Arrival
ToA	Time of Arrival
UDN	Ultra-dense Network
UE	User Equipment with 5G Connection
V2X	Vehicle to Everything Communication
VANET	Vehicular Ad Hoc Network

Symbols

Latin symbols

(a, b)	The input from UE in IQ algorithm
$add()$	Adding value to list Re
\mathcal{A}	The access structure
A_i	The coordinates matrix of AN_i
AN	The access node
\mathcal{B}	A binary representation of keywords
B_i	The distance matrix of AN_i
c	The input from A-UE in IQ algorithm
$cos()$	The cosine function
C	The cluster set i
$C^{-1}()$	The reverse of the cumulative distribution function
$e()$	The exponential function
d	The decryption key for Pub-pos and Pri-pos
d_i	The distance between UE and AN_i
d_c	The computed distance between UE and A-UE
d_m	The measured distance between UE and A-UE
$deq()$	The first value of the queue
\mathcal{D}	The survey dataset
\mathcal{D}	The location set for UE

Symbols

$DoA_{i,j}$	The extracted DoA between AN i and UE j
$E()$	The encryption of OPE
EN	The encrypted POIs in each cell
$EN()$	POI encryption with CP-ABE
\mathcal{F}	A randomized algorithm
g	The generator for CP-ABE
$H()$	The hash function
I	The query index
I_g	The grid index
I_k	The k-quadtrees index
ID	The ID of positioning signal
k	the number of k nearest POIs
k	The bit-length of random value
$keyword$	The raw keyword
$keyword_p$	The perturbed keyword
(k_a, r_a)	The OPE key generated by A-UE
(k_u, r_u)	The OPE key generated by UE
\mathcal{K}	The key generator from LBSP
K^B	The random generated key matrix
\mathcal{KS}	The acceptable keyword set
l	The dimension of random matrix RB
\mathcal{L}	The label generated by truth discovery module
\mathcal{LS}	The acceptable location set
m	The granularity of map
m	The number of POI in database
min	The threshold of neighbor numbers in DBSCAN
$max()$	The max value of the queue
$min()$	The min value of the queue

M	The total number of cells
\mathbf{M}	The POI-keyword matrix
MK	The master key of CP-ABE
Min	The min score of List Re
n	The number of points
\mathbf{n}	The number of keywords in database
\mathcal{N}	The number of random keys
P	The central point of cluster
p	The perturbation probability
PK	The public parameters of CP-ABE
Pos	The position list
\mathcal{PQ}	A priority queue
\mathbf{q}	The number of keywords in query
Q	The query from UE
\mathcal{Q}	LBS query from UE
\mathcal{Q}_t	The type list in LBS query
r	The random value generated by LBSP
r'	The random value generated by UE
R	The point list with distance to P less than ε
\mathcal{R}	The label generated by attack tracing module
RA_i	The random matrix for coordinate information of AN_i
RB_i	The random matrix for distance information of AN_i
Re	The recommended POI list
Re^g	The initial value of POI list Re
s	The dimension of random matrix RA
$sin()$	The sine function
$sort(,)$	Sorting function
\mathcal{S}	The response of IQ algorithm

Symbols

$Score$	The overall score of POI
$Score_{sp}$	The spatial score of POI
$Score_{tx}$	The textual score of POI
SK	The secret key of CP-ABE
t	The number of type in each query
T	The time period for each signal collection
\mathcal{T}	The trigger dataset
\mathcal{T}	A set of POI types
$ T $	The number of total type
$ToA_{i,j}$	The extracted ToA between AN i and UE j
UE	The end-user with 5G access
v	A random value in $[0,1)$
$W_{-1}()$	The Lambert function
(x, y)	The location of UE
(\bar{x}, \bar{y})	The coordinates of UE computed by FC
(x_a, y_a)	The location of A-UE
(x_i, y_i)	The coordinates of UE_i measured by AN
(x_p, y_p)	The perturbed location of UE
\mathcal{X}	The 3D coordinates of UE
z	A random integer

Greek symbols

ϵ	The privacy budget of LDP
ϵ_i	The privacy budget for UE in round i
ε	The threshold for neighbor distance
θ	A random number in $[0,2\pi)$
ϑ	The threshold of noise effect

1. Introduction

Positioning plays an important role in the fifth-generation mobile networks and wireless systems (5G). 5G positioning is a promising technology that can be used in many industrial and vertical applications cases, such as logistics, smart factories, autonomous vehicles, augmented and virtual reality, localised sensing, digital twins and so on. Compared with the traditional global positioning system (GPS), 5G positioning has advantages when it comes to accuracy, reliability and effectivity. 5G positioning can be realised through access nodes (AN), for example, base stations with ultra high density. Thus, it provides positioning with a wide coverage range, including in both indoor and outdoor situations [6, 114]. In addition, the high density of ANs increases line of sight (LoS) measures, greatly improving the accuracy of its positioning. Meanwhile, multiple LoS connections can further ensure positioning reliability. In contrast, GPS-based positioning is realised through GPS signals. However, these signals can be easily blocked or reflected by walls. As a result, it is impossible to calculate indoor locations because of low GPS signal strength. In addition, GPS-based positioning can be hacked because it is easy to simulate GPS signals using a strong power signal to mislead position calculation. Additionally, GPS positioning consumes a lot of battery power from the user equipment (UE) for signal receiving and processing; however, 5G positioning can save energy consumption on user equipment by migrating position calculations to a network-based platform such as an edge device or cloud server.

Despite these advantages, 5G positioning still faces many challenges, including trust, security and privacy. Mobile users expect positioning with privacy preservation and resistance to potential attacks. The network operator collects data and computes location information, thus ensuring the availability of a positioning service. Any malicious attacks and intrusions to a positioning service should be prevented. However, the network operator may outsource the positioning computation to another powerful party, which means the confidentiality of shared data and data processing should be ensured. The outsourced positioning tasks should be verified to prevent any dishonest positioning, which would send plausible positions without

performing any actual computation. In terms of location-based services, these should be re-evaluated to extract new requirements for 5G scenarios; for example, autonomous vehicles have strict latency requirements. In addition, not only the user protect their information from location-based service providers (LBSPs), but the data asset of the LBSP should also be protected, which is especially a concern giving the increasing value of service data and concept of 'pay-as-you-go'. The present dissertation aims to secure 5G positioning and its services by addressing these problems. Before exploring this, we first introduce 5G positioning and related services.

1.1 Background

Figure 1.1 illustrates 5G positioning and its services. There are four main stakeholders: location information service provider (LISP), LBSP, UE and location information collaborator (LIC). The role of each stakeholder is as follows:

- *Location Information Service Provider (LISP)*: It is normally a network service operator that provides the user positioning services. There are two main positioning models. In a network-centric positioning system, the LISP collects position measures from network devices such as base stations (BS) and executes the calculations either locally or on an outsourced computation server at the cloud/edge. The calculated position is sent back to the user through the cellular network. In user-centric positioning, the LISP transmits the relevant measurements to the user, enabling the user equipment to compute its position. The LISP also allows access to its database by third parties for location-based application development.
- *Location-Based Service Provider (LBSP)*: This is a platform that provides location-aware content based on user's position information. Some typical platforms include Google Maps, Uber, Wolt and so forth. To fulfil the services, the LBSP normally needs to create and maintain a certain database, which is one of the most valuable properties. When the user's request arrives, the LBSP processes the request based on their databases and sends location-aware content back to the user.
- *User Equipment with 5G Connection (UE)*: This refers to the user device equipped with broadband access through the 5G spectrum. The user device can obtain its position from the LISP through the 5G network or calculate its position using measurements from the LISP. It can also request the LBSP for certain location-based services.

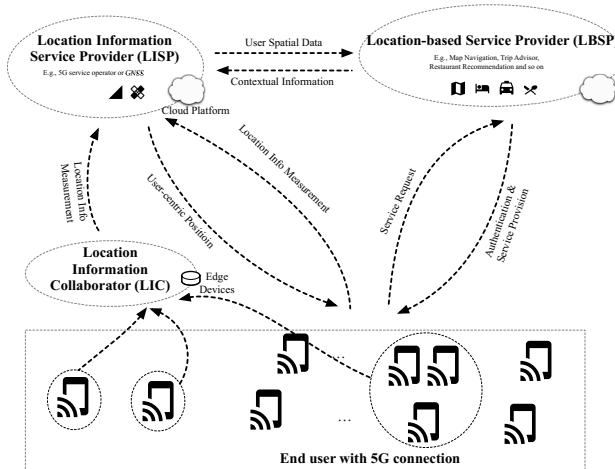


Figure 1.1. Illustration of 5G positioning and services

- *Location Information Collaborator (LIC)*: This refers to any mobile user in the network with whom the desired end-user can collaborate. Indeed, the 5G standard supports device-to-device (D2D) communications and collaborative communications, and such collaboration can also serve for positioning. One interesting application refers to crowd-positioning, where UEs are crowdworkers that offer positioning for another UE.

According to Fig. 1.1, the 5G positioning and services mainly include three parts: 5G positioning, verifiable positioning and location-based services (LBS). All these services are conducted collaboratively between multiple stakeholders. Detailed introductions of each are presented below.

1.1.1 5G Positioning

The key features of the 5G physical layer are an ultra-dense network (UDN) of access nodes (ANs) [43], large receiver bandwidths, massive multiple-inputs and multiple-outputs (MIMO) and device-centric architecture [74, 80]. Also, a 5G receiver should operate at high carrier frequencies, such as millimetre-wave communications ranging from 30 GHz to 300 GHz, and is always under LoS conditions with at least two transmitters or ANs, here because of the expected high density of the ANs. According to a 5G white paper [6, 20, 60], we consider the 5G positioning system as UDN, where ANs are attached to lamp posts with ten to hundred metres between each other, resulting in greatly increased LoS conditions between UEs with multiple ANs at a time [19]. Normally, each AN is equipped with an antenna array, allowing for estimating the direction of arrival (DoA) of a signal; AN locations are fixed and known.

The positioning process is illustrated in Fig. 1.2. UE transmits periodical

uplink pilot signals to ANs. Each AN selects only signals in an LoS condition by measuring a received signal strength (RSS) [66]. Based on the selected signals, the directional and temporal parameters, that is, the time of arrival (ToA) and DoA of the UE are extracted and delivered to a fusion centre (FC) that carries out the computation of the UE’s position. Note that the FC is a networked device with the capability of computation, storage and communication connection. It can be a roadside unit, a base station or even an outsourced cloud service provider. Eventually, the position information is sent back to UE through downlink beams or by the FC directly. The positioning computation can use trilateration, angulation or pattern matching methods.

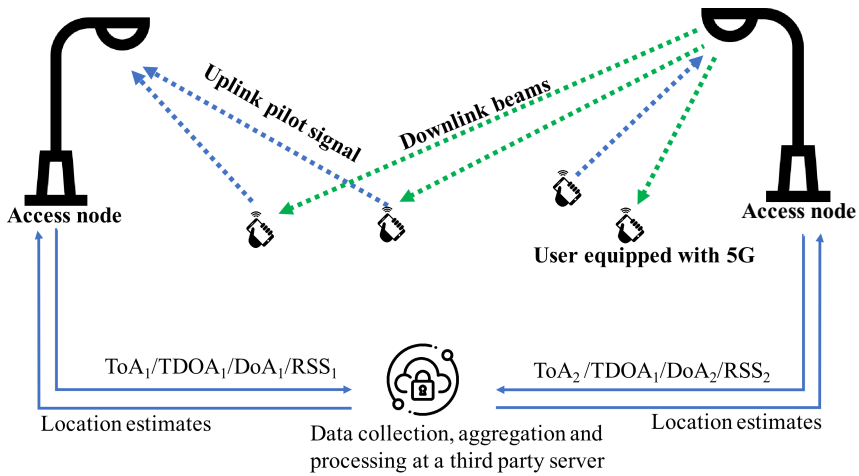


Figure 1.2. Illustration of a 5G UDN, where the access nodes are attached to lamp posts, and user equipment transmits periodical uplink pilot signals to the access nodes. The position estimation is carried out by a fusion center and sent back to the user equipment through downlink beams.

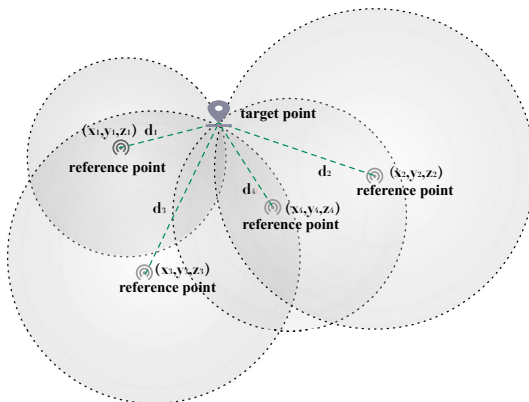


Figure 1.3. Example of trilateration-based 3D positioning

Trilateration

The ToA and time difference of arrival (TDoA) estimate the position using the trilateration method, which is based on the distance between the UE and ANs [28]. Trilateration finds the intersection of spheres, which are defined as a system of quadratic equations. Fig.1.3 is an example of 3D positioning based on trilateration, including a target point (normally UE) and at least four reference points (normally ANs) with known locations. These reference points are not in the same plane, and the target point is located in the range of these reference points. The sphere can be expressed as a quadratic equation according to Euclidean geometry, here based on the coordinates of the target point (x, y, z) , reference points (x_i, y_i, z_i) and distance d_i from the reference points to the target point. The target point communicates with the reference points nearby through enabled D2D links. During communication, the reference point can easily estimate its distance (d_i) to the target point through message travelling time. All four spheres are created similarly. The location of the target point can be obtained by finding the intersection point of four spheres, which, in theory, is equivalent to solving the independent linear equations, as shown in Equation 1.1.

$$\begin{cases} (x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2 = d_1^2 \\ (x - x_2)^2 + (y - y_2)^2 + (z - z_2)^2 = d_2^2 \\ (x - x_3)^2 + (y - y_3)^2 + (z - z_3)^2 = d_3^2 \\ (x - x_4)^2 + (y - y_4)^2 + (z - z_4)^2 = d_4^2 \end{cases} \quad (1.1)$$

Through multinomial expansion, Equation 1.1 can be expressed as a matrix computation as

$$AX = B, \quad (1.2)$$

where,

$$A = 2 \begin{bmatrix} (x_1 - x_2) & (y_1 - y_2) & (z_1 - z_2) \\ (x_1 - x_3) & (y_1 - y_3) & (z_1 - z_3) \\ (x_1 - x_4) & (y_1 - y_4) & (z_1 - z_4) \end{bmatrix}, X = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

$$B = \begin{bmatrix} d_2^2 - x_2^2 - y_2^2 - z_2^2 - d_1^2 + x_1^2 + y_1^2 + z_1^2 \\ d_3^2 - x_3^2 - y_3^2 - z_3^2 - d_1^2 + x_1^2 + y_1^2 + z_1^2 \\ d_4^2 - x_4^2 - y_4^2 - z_4^2 - d_1^2 + x_1^2 + y_1^2 + z_1^2 \end{bmatrix}$$

Thus, the intersection point in equation 1.1 can be revealed by one matrix inverse and one matrix multiplication as

$$X = A^{-1}B. \quad (1.3)$$

It is noteworthy that equation 1.3 is derived based on the assumption of 4 reference points. For the situation with more reference points (≥ 5) available, matrix A would not be invertible and its pseudoinverse should be used instead.

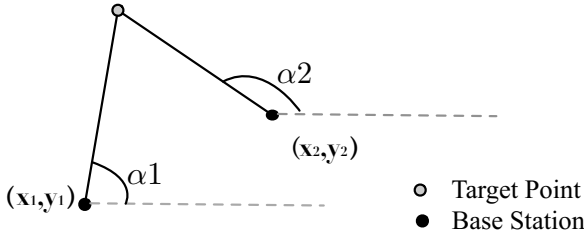


Figure 1.4. Example of angulation-based positioning

Angulation

The DoA is estimated using the angulation method. In contrast to trilateration, the measurements here are the angles between the target and multiple reference points [59]. To derive these angles, these reference points must be equipped with antenna arrays and reveal their coordinates information.

The basic principle behind angulation is illustrated in Fig. 1.4. The angle of an incoming pilot signal is measured at the reference point, thus restricting the target's position along a line that intersects both the target's and reference point's position. If the angle to a second reference point is taken into account, another line is defined and the intersection of both lines then represent the target's position. Thus, from a theoretical point of view, it is sufficient to generate angle measurements with two reference points to obtain a 2D position and three reference points for a 3D position.

Pattern match

Pattern match, which is also known as fingerprint-based positioning, consists of offline and online procedures. Fig. 1.5 is an example of a pattern match based on RSS. The offline procedure is the training phase, which is responsible for database construction, data preprocessing and model training. The database is normally composed of all the collected RSS information generated by the ANs deployed at the positioning service area. These data are preprocessed into structured training data with operations such as feature extraction, regularization and so forth. During model training, machine learning or deep learning is often applied to extract the core features of the training data. kNN [67], support vector machines (SVMs) [29] and neural networks (NN) [34, 100] are all popular models that have been applied for this problem. The trained model is then employed to provide a prediction service for a target UE with the input being the real-time RSS data received by the UE.

In conclusion, a variety of positioning mechanisms can be adopted here to calculate the position. The positioning mechanisms are summarised in Table 1.1.

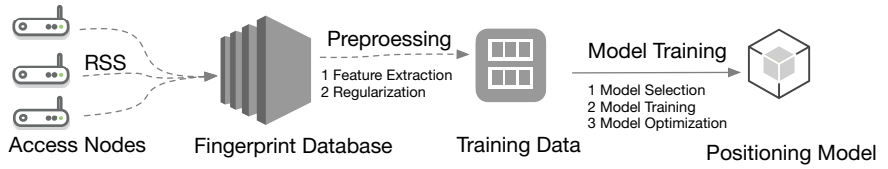


Figure 1.5. Example of RSS-pattern-match-based positioning

Table 1.1. Summary of the positioning mechanisms

Method	Trilateration	Angulation	Pattern match
Mechanism	ToA & TDoA	DoA	RSS
Illustration	Fig. 1.3	Fig. 1.4	Fig. 1.5

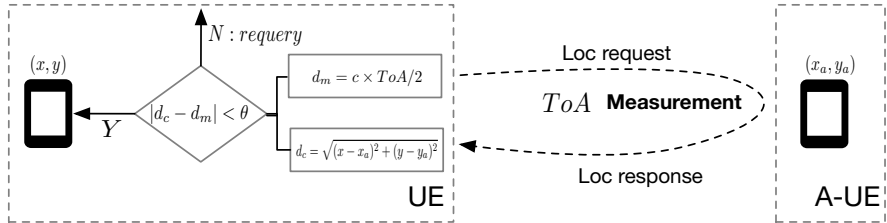


Figure 1.6. Overview of D2D cooperative location verification

1.1.2 Verifiable Positioning

The appearance of 5G enabler technologies like D2D, V2X and cloud/edge computing has encouraged the development of new positioning systems. However, threats in these emerging positioning systems are still unresolved. Verification is an effective method to preserve the integrity of new positioning systems, especially in outsourced environments.

Verification of outsourced positioning model

By offloading the data processing to a network edge closer to users, an edge computing-based positioning service offers the advantages of lower latency and relieved server stress compared with a centralised model. However, one crucial problem is that the edge device running the outsourced services may be untrusted. This device may cheat the LISP by providing fake results without following the designed protocol or even maliciously outputting some poisonous results to mislead users. Thus, research to verify the correctness of data processing at the edge to enhance its trustworthiness, especially for location-related services, is urgently needed.

D2D cooperative location verification

D2D cooperative location verification has been extensively used in scenarios such as vehicular ad hoc networks (VANETs) and indoor positioning. Despite its numerous applications, the general idea is to verify the distance using a nearby device, which is done by comparing the measured distance with the computed distance based on known locations. Fig. 1.6 presents an overview of D2D cooperative location verification, which happens between UE and assistant user equipment (A-UE). To do this, UE initiates a request to a nearby A-UE for its location. Once the A-UE accepts the request, it immediately responds with its location to the UE, which can easily calculate its distance (d_c) based on these two locations. At the same time, it also measures the distance between them using the time difference (ToA) between the query and response ($d_m = c \times ToA/2$), here by assuming the signal travels at the speed of light c . If the difference between d_c and d_m is under a threshold θ , which is influenced by signal noise and request processing time, the verification is successfully completed with success. Otherwise, the location should be recalculated.

1.1.3 Location-based Services

LBS has brought a lot of convenience to our lives. It provides a tailored information service, for example, restaurant recommendations based on a provided location. The popularity of smartphones enables the collecting and sharing of various types of data in an open and public way, for example, pictures, videos, locations, and so on. Based on these data, the LBSP maintains a POI database and provides a POI recommendation service to UEs. A UE sends its query to the LBSP to obtain POI recommendations. In essence, the LBSP collects the POI from the public and maintains the database to provide the POI recommendation service to UEs.

A LBS query model is shown in Fig 1.7. Let D be the POI database, which is defined as a list composed of spatial coordinates (x, y) where (x, y) corresponds to the longitude and latitude of POI and a key value set (k, c) , where k is the related keyword and c is its frequency counting. We suppose that there are m POIs and n keywords in the database. The end-user sends its query to get the POI recommendations from the LBSP. A query Q is defined similarly with the query location (x, y) and the query keyword list $\{k_1, k_2, \dots, k_q\}$. The keyword-enabled top- k POI recommendation is used to find the top k POIs with the highest relevance scores to query Q ; the formal definition is given as follows:

Definition 1 (Keyword-Enabled Top- k POI Recommendation). Given a query Q and a POI dataset D , finding k POIs in D with the highest relevance scores.

The ranking function [112] that computes the relevance score of a query

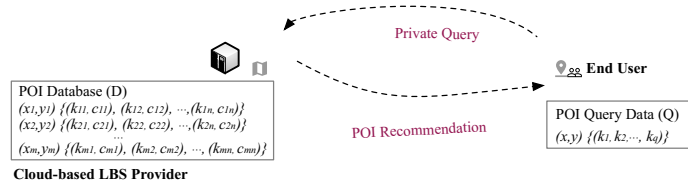


Figure 1.7. Overview of LBS query

Q and a POI record D_i is defined as

$$Score(Q, D_i) = \alpha Score_{sp} + (1 - \alpha) Score_{tx},$$

where $Score_{sp}$ and $Score_{tx}$ are the spatial relevance score and the textual relevance score between Q and D_i , respectively. The value α determines the importance of the spatial or textual relevance score in the overall score. Concretely, $Score_{sp}(Q, D_i)$ is defined as follows:

$$Score_{sp}(Q, D_i) = 1 - \frac{Euc(Q.loc, D_i.loc)}{Euc_{max}}, \quad (1.4)$$

where $Euc(Q.loc, D_i.loc)$ is the Euclidean distance between Q and D_i , and Euc_{max} is the maximum possible Euclidean distance between any points in the space under consideration. For $Score_{tx}(Q, D_i)$, it can be computed as follows:

$$Score_{tx}(Q, D_i) = \frac{\sum_{D_{i,k} \in Q.k} D_{i,k}(c)}{\sum_{D_{i,k}} D_{i,k}(c)}, \quad (1.5)$$

which is a fraction of the keywords in query Q among the keywords in D_i .

1.2 Security and Privacy of 5G Positioning and Services

This section elaborates on the security threats and potential privacy leakage in 5G positioning and their related services. On the one hand, 5G positioning and services provide a lot of benefits to both the network operators and end-users, for example, through its potential to enhance location-aware communications and intelligent transportation. On the other hand, 5G positioning comes with increased privacy concerns from the participant's point of view because it has been proven to be sensitive to intentional interference and security breaches during positioning [53]. Moreover, with the advent of cloud and edge computing, LISP likely faces challenges from these untrusted computation centres.

1.2.1 Security Threats

Security threats are the vulnerabilities related to the reliability and integrity of positioning in the presence of interference, attacks or unintentional errors. The current dissertation includes four common security

attacks on 5G positioning and its services. These attacks are ranked from weakest to strongest according to their attack ability.

- **Eavesdropping:** this is a passive attack where the attacker passively monitors the network communications to capture communication data.
- **Jamming attack:** this is an active attack where the attacker tries to degrade the positioning accuracy by adding noise to positioning signals. Different from the spoofing attack below, in this case, the attacker is unable to manipulate the signals by deleting, changing or reordering them.
- **Spoofing attack:** this active attack is where an attacker intercepts the communication path or positioning signalling between two parties, thereby obtaining credential data or performing message modification by deleting, adding to, changing or reordering the message. The message can be, for example, the positioning signalling message between the LISP and UE.
- **Collusion attack:** in this attack, an attacker can hack or even set up fake ANs in 5G networks and send incorrect signalling messages to deteriorate the positioning system. The attackers here also have access to credential information such as static keys.

1.2.2 Privacy Leakage

Privacy leakage refers to the disclosure of sensitive data such as participants' data, location and identity information. The risks of losing one's privacy can range from mild discomforts to serious dangers of burglary, theft or even loss of life. A list of privacy leakage considered in 5G positioning and services is as follows:

- **Potential misuse of personal information:** when UEs agree to the distribution terms of the LBS and localization engines on their mobile device, the terms are sometimes not very clear or too broad, hence leading to the potential legal misuse of user personal data. For example, based on the location-related query from a UE, the LBSP or LISP can infer significant personal information, such as work and home places, attended schools, health problems and so on.
- **Unauthorised tracking:** apart from revealing sensitive information like work and home addresses, the LISP and LBSP may conduct unauthorised monitoring of the user's movement when they have continued access to the user's location information.

- **Right to be forgotten:** the storage of personal data at the LBSP and LISP is often not transparent to users. The right to be forgotten rule in Europe states that a person has the right to ask for the removal of their data when such data are no longer necessary for the purposes for which it had been initially collected.
- **Pay-as-you-use:** this is a payment model in cloud computing that charges based on resource usage. For service providers such as the LBSP or LISP, a UE is only eligible to obtain the information related to its query. For example, a UE only obtains k recommendations from a k NN query.

In 5G positioning, different stakeholders have different concerns about security and privacy. Mobile users expect positioning with privacy preservation and resistance to any potential attacks. Network operators collect data and compute location information; thus, the availability of positioning services are its focus. This means that any malicious attacks and intrusions to the positioning service should be prevented. On the other hand, the network operator may outsource the positioning computation to another party, which means the confidentiality of shared data and data processing should be ensured. In terms of location-based service providers, these providers should not know the exact location information of users. Based on the security and privacy research of 5G positioning and services, we present the research problems addressed in the current dissertation.

1.3 Research Problems

The ultra high density of ANs, large receiver bandwidth and network-based computation enable 5G positioning to achieve below one-metre accuracy with low energy consumption on mobile devices. However, these new characteristics of 5G positioning propose threats to users. The present dissertation aims to investigate trustworthy and privacy-preserving 5G positioning to help end-users, network operators and location-based service providers. Focusing on 5G positioning and its services, we propose the following research problems.

1.3.1 5G Positioning

5G positioning aims to offer highly precise position information to mobile users and supports massive commercial mobile services. However, the gap between its theory and practical usage is still large because of several security and privacy reasons.

Security of 5G Positioning

First, most existing methods have been proposed based on the availability of clean data [99], which, however, is not a normal situation in practice because all kinds of erroneous data inputs could be injected by broken ANs or even malicious attackers [53, 82]. Thus, these methods may not work in practice as shown in lab simulations. Kai et al. [35] and Manesh et al. [56] both implemented systems with GPS spoofing attack detection and attacker localisation. Singh et al. [87] designed the first secure ranging system resilient to distance enlargement and reduction attacks. Abdalla et al. [1] studied the attacks and corresponding mitigation strategies of 5G networks by leveraging UAVs.

As we can see, although many efforts have been devoted to attack detection and defence, they are designed on a case-by-case basis and are not suitable for solving 5G positioning attacks. The difficulty of attack detection resides in the complexity of the 5G positioning environment, where the noise can be erroneous data from broken ANs, system noise from clock synchronisation, spoofed signals from malicious parties and obstacle signals, which are known as non line of sight (NLOS) and are caused by physical objects like buildings or trees. Studies have developed effective solutions like maximum likelihood, least squares and constrained optimisations, as well as robust statistics for NLOS identification [58, 71]; however, there is still lacking sufficient identification techniques targeting the other noise data stated above, especially malicious attacks on 5G positioning.

Thus, the key research problems for enhancing the security of 5G positioning can be summarised as follows: (1) How can we ensure data truth to enhance positioning accuracy? (2) How can we effectively distinguish different attacks? (3) How can we discover potential attackers who provide wrong or malicious data? By solving these research problems, we aim to develop a holistic noise data identification and truth discovery method to spur the further development of 5G positioning in practice.

Privacy of 5G Positioning

Second, because of the confidentiality of personal positioning information, preventing information leakage is another key issue in 5G positioning research. Different from traditional station-based or GPS-based positioning, 5G positioning, especially cloud/edge-based 5G positioning, is characterised by network-based positioning capabilities, which creates more threats to mobile users, especially regarding privacy. In addition, with its further applications in location-sensitive fields such as autonomous driving, spatial crowdsourcing and so on, it is crucially important to provide reliable and secure positioning with privacy preservation that would be applicable in different scenarios.

In the context of 5G positioning, privacy becomes a crucial issue when the positioning computation is outsourced to third parties. Current techniques for privacy enhancement in positioning fall into physical-layer-based privacy enhancement and application-driven cryptographic solutions [53]. At the physical layer, reliability monitoring and outlier detection algorithms are absorbed to detect and locate interference signals [27]. At the application layer, which is one focus of the present dissertation, cryptographic techniques are integrated against malicious attacks with four primary goals: confidentiality, integrity, authenticity and nonrepudiation [101]. Although positioning information can be well preserved using cryptography [33, 41, 105], a limitation still exists regarding computational complexity, low efficiency, and poor feasibility.

Thereby, the main challenge here is as follows: (1) How can we effectively and efficiently preserve the privacy of both the target and reference points if the positioning computation is outsourced to third parties? A delicate and innovative design is expected to overcome these limits and balance the trade-off between privacy, security and efficiency, especially for satisfying the fast response requirement in 5G positioning.

1.3.2 Verifiable Positioning

One growing tendency in 5G is to outsource computation tasks to a more powerful computation server, such as edge nodes. The benefits of outsourcing are threefold: First, it greatly relieves the computation burden of UE; Second, it improves the user experience by deploying the service to edge points close to the users, thus reducing the message transmission time and computational overhead; Third, the distribution model of edge computing can prevent potential bottleneck attacks. Despite all the benefits, there are still some open issues when it comes to positioning outsourcing. One is how to ensure that the outsourced positioning service is executed honestly by the computation server without any erroneous or malicious tampering with the results. A dishonest computation server can modify the outsourced positioning tasks to return plausible results without performing the actual work. Considerable attention has been devoted to this issue, and extensive solutions have been proposed to address the verifiable problems for matrix factorisation [22, 98], keyword search [115], inner product evaluation [52], including the use of secure processors [88], trusted platform modules [42], interactive proofs [26] and probabilistically checkable proofs [8]. Despite their effectiveness, these verification solutions either depend on secure processors or trusted platform modules, have heavy overhead costs [25] or assume cooperation between noncolluding trusted parties [113]. Because of these limitations, they are not suitable for outsourced 5G positioning scenarios that have limited computation and storage resources at the edge

nodes.

Another advantage of 5G is D2D communication, which allows a device to send messages to devices in proximity without going through the core network. It is especially handy for position verification because the device can verify its position with nearby devices through D2D communication channels [39]. Although D2D cooperative location verification simplifies the verification process substantially, the disclosure of location information during verification raises a big privacy risk for participating assistant devices because they must send their real-time locations to another device while the holder of this device is unknown and could be malicious. Research has shown that 46% of teen users and 35% of adults turn off location sharing because of privacy concerns [76]. According to the adopted strategies, the privacy protection solutions can be classified into three categories: access control, obfuscation and cryptography. Access control achieves privacy protection by filtering unauthorised participants, and obfuscation disguises real information with randomly generalised fake information. However, they normally depend on a central gateway for data filtering or obfuscation, making them unsuitable for decentralised D2D scenarios [10, 17, 61]. In contrast, solutions relying on cryptography offer high-level privacy and security guarantees while being independent of a central gateway. In the literature, Paillier encryption and garbled circuits are two popular cryptographic tools that have been applied to this problem. Based on their inherent properties, these two cryptographic tools can implement algebraic computing in ciphertext so that sensitive information can be protected while the service runs smoothly. However, the practical implementation of these two solutions is prohibitive because of their high overhead. Herein, a privacy-preserving, lightweight and efficient solution appropriate for decentralised D2D scenarios is urgently needed.

In conclusion, the key research problems of verifiable positioning are as follows: (1) How can we ensure the integrity of outsourced positioning services and their honest execution at the edge? (2) How can we achieve a solution that considers the limited computation and communication storage at the edge and preserves positioning services from being influenced by verification? (3) How can we protect data privacy in distributed D2D location verification while meeting the low-latency requirement of 5G positioning?

1.3.3 Location-Based Services

With the prevalence of smartphones, LBS has been developed to provide tailored information services to users according to their location-related requests. Furthermore, public awareness of privacy has generated a lot of research about privacy-preserving LBS. On the one hand, the user

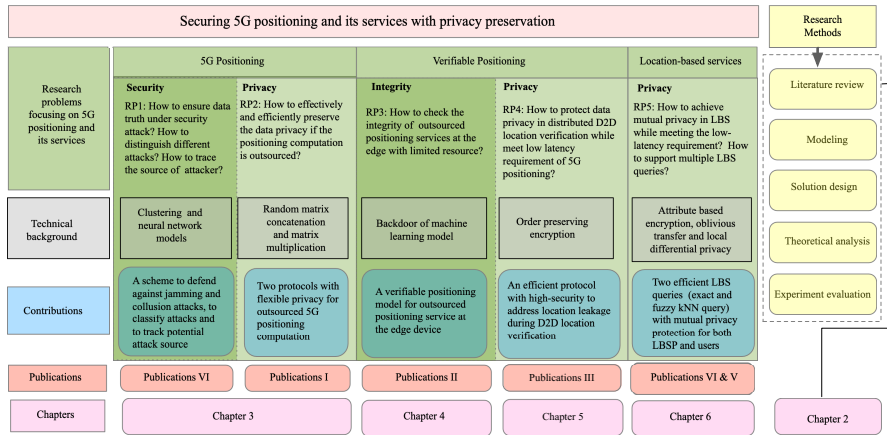


Figure 1.8. The relationship among research problems, technical background, contributions, publications and chapters

requires their query data, especially position information to be protected because the disclosure of such sensitive information could threaten their properties and even life. On the other hand, the LBSP wants to protect its collected database from any unauthorised parties, restricting free access to it because the database is considered a valuable asset of the LBSP, one that requires an enormous investment to collect and maintain.

Solutions such as mix zone [12, 23] k-anonymity [10, 17, 61] and dummy location [40, 55] were first provided to prevent the LBSP from distinguishing the request user. However, these tactics neglect mutual privacy, failing to consider the protection of the LBSP's data asset [51]. Cryptographic techniques such as homomorphic encryption [69, 107] and private information retrieval (PIR) [24, 70] were then introduced to solve this problem. Despite the mutual privacy they manage to provide, they also incur heavy overhead, which is prohibitive in practice. Nevertheless, overhead caused by privacy protection is not the only reason for this high latency. Traditional LBS systems normally maintain a centralised processing scheme with LBS remotely deployed in the cloud [37]; this centralised scheme creates high latency for all users. Low latency is crucial for LBS queries considering the fields these queries are used within, for example, internet of vehicles (IoV), automated driving and so on.

To address the above problems and practically deploy LBS into critical 5G scenarios, the research problems focus on the following: (1) How can we efficiently preserve the mutual privacy of the user and LBSP during LBS provision? (2) How can we support multiple query schemes? (3) How can we meet the low-latency requirement and fit it into the distributed 5G LBS scenario? The proposed solutions are expected to further stimulate the development of LBS applications in location-sensitive scenarios.

1.4 Dissertation Work Introduction and Contributions

The current dissertation aims to secure 5G positioning and its services using privacy preservation, which is done by solving the above research problems. The relationship among the research problems, research contributions, included publications and chapters are summarised in Fig. 1.8.

Specifically, the current dissertation applies the following research method to achieve its research objectives: (1) study the specific scientific research problems of each research objective through a literature review; (2) explore the theoretical basis and technical methods for solving the specific research problems, such as machine/deep learning, trusted computing, clustering methods, backdoor, differential privacy and so forth; (3) propose novel solutions to the problems based on selected theories and technologies; (4) develop performance evaluation metrics and employ them to demonstrate, analyse and validate the performance and effectiveness of the proposed solutions through theoretical proof, simulation experiments and real-world data tests, while showing their advantages by comparing them with existing works; (5) and further improve our solutions and deploy them in prototypes while striving for publication and technology transfer to fully achieve our research goals.

The concrete research work and contributions involving 5G positioning and its services are introduced below.

1.4.1 Security and Privacy in 5G Positioning

The research on 5G positioning includes both security and privacy aspects. The security research of 5G positioning aims to enhance the positioning accuracy using truth discovery, potential attack detection and attack tracing under jamming and collusion attacks. To mitigate the jamming and collusion attacks in a network-based 5G positioning system, a novel scheme composed of three modules is proposed. A truth discovery module applies a clustering-based method aiming to generate the most approximate position value and find suspicious signals. Based on neural network models, we further develop an attack detection module and attack tracing module to perceive the attack category and locate malicious sources. Through simulation, we conduct extensive experiments to illustrate the effectiveness of our scheme. The results show an average of 1.2 metres positioning accuracy with the truth discovery module, 86% attack detection accuracy and 82.5% attack tracing accuracy. The results also show high detection and tracing accuracy with very simple neural network models (one hidden layer with 20 neural units). The simplicity of our scheme implies the potential for practical deployment. The related work is published in Publication VI and presented in Chapter 3.

The privacy of 5G positioning focuses on information protection in outsourced positioning computations. In outsourced computations, the location of the reference points and their distance to a target point are sent to third parties. However, these data are quite sensitive because they can be easily used to locate the reference points once exposed. The potential information disclosure when outsourcing computations could decrease the motivation of public participants to provide positioning assistance as reference points. The current dissertation presents two efficient protocols to address the above research problem. By leveraging matrix concatenation and multiplication, these protocols can disguise the original sensitive data, including both distance and location information, into a random matrix while keeping the positioning result intact. Compared with existing work, our proposed protocols have shown significant efficiency improvement. The results show that our proposed protocols are at least 4.5 times faster when compared with other work in terms of positioning service provision. The related work is published in Publication I and presented in Chapter 3.

1.4.2 Verifiable Positioning

The emergence of 5G enabler technologies like D2D, Vehicle to Everything (V2X) communications, cloud/edge computing have led to the development of new positioning systems [31, 68]. These new positioning systems facilitate the positioning service in different scenarios. However, several threats in these new positioning systems still remain [94, 103, 109]. This part of the research can be divided into two areas: (1) For positioning services outsourced to edge devices, how can we ensure the integrity of these outsourced positioning services? (2) For location verification between UEs, how can we preserve the privacy of the participants without blocking services? Apart from the privacy requirement, the challenge of solving these problems is in improving efficiency while preserving service accuracy.

We solve the first problem by introducing a 'backdoor' concept. The backdoor used to be a weakness in machine learning because it allows adversary injection attacks. Nevertheless, inspired by this idea, we designed a verification scheme for an edge-based positioning system to preserve the integrity of outsourced positioning services at untrusted edge points. The idea is to inject a specially designed private dataset into the offline training of the positioning model before outsourcing it to edge devices. Then, the integrity of the outsourced model can be verified by the LISP through the injected dataset. The verification is successful only when the prediction accuracy can pass a certain threshold. With our scheme, the LISP can easily check if the edge point is honestly running the outsourced service or not without any extra alterations to the existing system. To the best of our knowledge, this is the first work aiming to achieve verifiable positioning through a 'backdoor' strategy. Different from traditional solutions like

the use of secure coprocessors or trusted platform modules, this scheme is efficient and independent of any third party or product. The effectiveness of the proposed scheme is verified through extensive experiments using state-of-the-art positioning models on real-world datasets. The related work is published in Publication II and presented in Chapter 4.

Different from the first problem that focuses on verifiable positioning in outsourced computations, the second open problem is proposed in D2D-based cooperative location verification. D2D provides an easy solution for users to check the integrity of their location information using nearby assistant devices. However, doing so requires the nearby assistant devices to reveal their real-time location to the user, putting all the assistant devices at risk. The current dissertation has aimed to provide an efficient solution with a high-level privacy guarantee. Based on order-preserving encryption (OPE), we propose an efficient protocol with a high-security guarantee to address this issue. Specifically, we propose an innovative coordinate-based verification method and efficient privacy-preserving protocol for interval queries based on order-preserving encryption. Privacy-preserving location verification can be easily achieved by applying the designed interval query to the coordinates, here based on a verification method. The experiment is conducted in comparison with two state-of-the-art solutions implemented with Paillier and garbled circuits, respectively. The results show that the proposed solution outperforms existing work in terms of online latency. Under the same security level, the proposed solution can complete verification under 30 ms, while the Paillier and garbled circuit-based solutions take 150 ms and 100 ms, respectively. The related work is published in Publication III and presented in Chapter 5.

1.4.3 Privacy-Preserving LBS

LBS has drawn intense attention as the most important positioning application. Meanwhile, public awareness of privacy has led to the research on privacy-preserving LBS. The challenge in privacy-preserving LBS relies on three areas: (1) How can we achieve the mutual privacy of both the LBSP and UEs? (2) How can we support a variety of LBS queries? (3) How can we meet the low-latency requirement of 5G? (4) How to quantify the value of privacy?

K nearest neighbors (kNN) is selected as the representative LBS points of interest (POIs) query because it can be easily adapted to other LBS queries such as nearest query, range query and so on. Apart from the traditional exact query, the present dissertation also supports fuzzy queries, which reply with approximate answers. As an example, for a user searching for the nearest restaurant, the exact query replies with the restaurant geographically nearest to the user, while the fuzzy query replies with any restaurant rather than exactly. Compared with the exact query, a fuzzy

query maintains the response quality while achieving a faster response. Additionally, the support of filtering based on POI types should also be considered.

The current dissertation has implemented privacy protocols for both exact kNN and fuzzy kNN queries with keyword constraints. The exact kNN query is composed of the system initialisation and online query. The system initialisation is processed offline, where the LBSP encrypts its POI data and distributes the encrypted data to each fog node according to their coverage regions. The online query is performed based on GPS coordinates and optional POI types. During the online query, a user obtains a private key from the LBSP and then performs decryption over the encrypted POIs retrieved from a fog node. Technically, we realise data encryption with ciphertext-policy attribute-based encryption (CP-ABE) and privacy-preserving key retrieval with oblivious transfer (OT). The fuzzy kNN achieves query privacy using random perturbation. Based on local differential privacy (LDP), geo-indistinguishability is introduced for the perturbation of location in the query, and text indistinguishability is used for the perturbation of keywords in the query. After perturbation, the LBSP is unable to distinguish the perturbed query from the original one, with a probability of more than e^ϵ ; Here, ϵ is used to measure the privacy. Generally, a smaller ϵ implies a better privacy protection. Based on the proposed solutions, the privacy of the user's query can be protected while allowing for approximate searching. Significant analysis and empirical study based on real-world datasets have been conducted to prove the superiority of our protocols. The experimental results show an average of 2000 ms latency for an exact kNN query and a 500 ms latency for a fuzzy kNN query of over 100 queries. The detailed work is published in Publication IV, Publication V and concluded in Chapter 6.

1.5 Dissertation Structure

The overall dissertation consists of an introductory part with six chapters appended by six publications. The remainder of this dissertation is organised as follows:

Chapter 2 presents a preliminary introduction to the technologies adopted in the current dissertation. It also reviews the literature work for each research focus.

Chapter 3 focuses on security and privacy problems in 5G positioning. A novel scheme composed of truth discovery, attack detection and tracing is proposed to mitigate the jamming and collusion attacks in a 5G positioning system. Two efficient protocols (Pri-pos and Pub-pos) are introduced to address the potential privacy leakage in 5G positioning. Theoretical analysis and performance evaluation based on the proposed work is conducted and

discussed. In the end, we conclude the chapter with a summary.

Chapter 4 investigates the problem of the verifiable positioning model. It presents the training and verification of the 'backdoor'-based positioning model. Extensive experiments based on the real-world dataset are conducted to prove the effectiveness of the proposed solution. The chapter ends with a summary.

Chapter 5 illustrates the design of a privacy-protected solution for D2D cooperative location verification. The experiment is compared with Pailler and garbled circuits solutions, and this shows the efficiency of our proposed solution. Similarly, this chapter is composed of a problem statement, solution design, experimental evaluation and summary.

Chapter 6 exhibits the design of two privacy-preserving LBS queries (exact kNN and fuzzy kNN query) that achieve mutual privacy and high efficiency. The security of both queries is thoroughly analyzed. Experiments based on the real-world dataset are conducted. The results show that our solutions outperform the existing work, showing an advantage in terms of online latency. The chapter ends with a summary.

Finally, Chapter 7 concludes the compendium with a discussion of current work and future perspectives. Particularly, it concludes with the main contribution, applicability and limitation of the presented work, suggesting future research directions.

2. Preliminaries and Related Work

This chapter introduces the technical background of this dissertation. Additionally, this chapter also reviews and compares the related work regarding 5G positioning and its services.

2.1 Preliminaries

2.1.1 Order-Preserving Encryption

OPE is a symmetric-key algorithm with an order-preserving property, in which the orders of ciphertexts are the same as those of their plaintexts [4, 57]. This property enables the sorting and ranking of encrypted data without revealing plaintexts. The current paper adopts the OPE implementation proposed in [38]. This method encrypts the values with linear computation so that the values can be disguised while the order of the values can still be preserved. It consists of three main parts: key generation, encryption and decryption:

Key generation: It generates a key pair (k, r) , where k and r are both random integers with $k \in \mathbb{Z}$ and $0 \leq r < k$. As a symmetric-key algorithm, the key is used for both encryption and decryption.

Encryption: For the value m , the encryption function is defined as $E(m) = k \times m + r$. The security of the OPE is defined with indistinguishability under ordered chosen-plaintext attacks; hence an adversary cannot deduce any information about the plaintexts, except for the order when the ciphertext space is at least three times bigger than the plaintext space [38].

Decryption: For ciphertext c , the decryption function is defined as $D(c) = (c - r)/k$.

2.1.2 Paillier Cryptosystem

Paillier cryptosystem [2, 69] is a probabilistic asymmetric algorithm for public key cryptography. The encryption function is defined as $E_{pk}(m, r) = g^m \times r^N \pmod{N^2}$ where $m \in \mathbb{Z}_N^*$ is a message for encryption, N is the product of two large prime numbers p and q , g generates a subgroup of order N , and r is a random number in \mathbb{Z}_{N^2} . The public key for encryption is (N, g) and the secret key for decryption is (p, q) . The details of decryption function D with secret key sk can be found in [69]. Paillier cryptosystem owns semantic security. Given a set of ciphertexts, an adversary cannot deduce any information about the plaintexts with Paillier encryption. Another notable feature of the Paillier cryptosystem is the homomorphic property. The product of two ciphertexts will be decrypted to the sum of their corresponding plaintexts, and the k^{th} power of a ciphertext will be decrypted to the product of k and its corresponding plaintext.

2.1.3 Garbled Circuits

Garbled circuits proposed by Yao [106] allow two semi-honest parties holding inputs x and y , respectively, to evaluate an arbitrary function $f(x, y)$ without leaking any information about the inputs beyond what can be deduced by the function output. The solution is that one party (the garbled-circuit *constructor*) constructs a garbled version of a circuit to compute f , while the other party (the garbled-circuit *evaluator*) obviously computes the output of the circuit without learning any intermediate values [32]. Three simple circuits (the Comparison, ADD and OR) will be used in this dissertation to realize secure location verification in Chapter 5. A Comparison circuit takes two σ -bit integers x and y as input and outputs 1 if $x > y$ and 0 otherwise. ADD circuit takes two σ -bit integers x and y as input, and outputs a $(\sigma + 1)$ -bit integer z , such that $z = x + y$. An OR circuit takes two σ -bit integers x and y as input, and outputs a (σ) -bit integer z , such that $z = x \vee y$, where \vee is the bitwise OR operation. The details of Comparison, ADD and OR circuits can be found in [44].

2.1.4 Local Differential Privacy

The LDP assumes that an adversary has access to the personal response of an individual in the database. In the LDP, each data owner must perturb its data locally using a randomised mechanism before sending it to a collector. Nevertheless, the statistical estimation error on the perturbed data is as small as that of the raw data. This technique is especially useful for distributed data collection.

Formally, let D denote the whole database and \mathcal{A} be a randomized algorithm that takes a data tuple t as the input and outputs t^* . ϵ -local

differential privacy is defined on \mathcal{A} and a privacy budget $\epsilon > 0$, as follows:

Definition 2 (ϵ -local differential privacy). An algorithm \mathcal{A} satisfies ϵ -local differential privacy, iff for any input tuples $t, t' \in D$ and for any output t^* , the following inequality always holds:

$$Pr[\mathcal{A}(t) = t^*] \leq e^\epsilon \times Pr[\mathcal{A}(t') = t^*].$$

ϵ -LDP means that by observing the output t^* , the data collector cannot infer whether the input tuple is t or t' with a confidence higher than e^ϵ . The LDP also has the sequential and parallel composition property that can be used to facilitate modular design and analysis. The sequential composition guarantees that the overall LDP privacy for algorithms operated in a sequence is equal to the sum of their privacy budgets. The parallel composition promises that the overall LDP privacy for algorithms operated in parallel is the max of their privacy budget. A formal definition is given below.

Theorem 1. (*Sequential Composition*) *If we have n numbers of differentially private mechanisms $\mathcal{A}_1, \mathcal{A}_2, \dots, \mathcal{A}_n$, each providing ϵ_i -local differential privacy, then any sequence composition of these mechanisms that yields a new mechanism \mathcal{A} is $(\sum \epsilon_i)$ -local differential private.*

Theorem 2. (*Parallel Composition*) *If we have n numbers of differentially private mechanisms $\mathcal{A}_1, \mathcal{A}_2, \dots, \mathcal{A}_n$, with each providing ϵ_i -local differential privacy, then any parallel composition of these mechanisms that yields a new mechanism \mathcal{A} is $\max(\epsilon_i)$ -local differential private.*

The randomised response has been the predominant perturbation mechanism for LDP. For sensitive boolean data, it replaces the genuine data with probability p and gives the opposite data with probability $1 - p$. To satisfy ϵ -LDP, p can be set as follows:

$$p = \frac{e^\epsilon}{1 + e^\epsilon}.$$

Apart from boolean data, it can also be generalised to a more complex problem by converting the input data into a binary form and applying a perturbation to it.

2.2 Related Work

The related work includes the discussion of security and privacy in 5G positioning, verifiable positioning, privacy protection in D2D location verification and privacy-preserving LBS.

Table 2.1. Related work comparison w.r.t. 5G positioning security

Ref.	Robustness	Att. reduction	Att. detection	Att. classification	Att. tracing	Performance
[35]	Spoofing attack		✓		✓	High
[56]	Spoofing attack	✓				High
[95]	Spoofing attack		✓			High
[87]	Distance enlargement and reduction attack	✓				Medium
PVI	Jamming and collusion attacks	✓	✓	✓	✓	High

2.2.1 Security and Privacy in 5G Positioning

Security in 5G Positioning

The security research aims to improve the reliability and integrity of the 5G positioning system in the presence of interference, attacks and unintentional errors. In Table 2.1, we evaluate the related work regarding robustness, functions and efficiency:

- Robustness refers to the ability to resist specified attacks.
- Attack reduction, detection, classification and tracing are the supported solutions that are used to mitigate the influence of attacks. Specifically, attack reduction reduces the influence of an attack. Attack detection and classification indicate whether the solution can find an attack and classify its type, while attack tracing refers to the ability to report the source of attacked signals.
- Performance is the measurement of efficiency. It is divided into three levels based on their operation time when compared with raw methods (methods without any security protection). High means the operation time of the implemented solution is similar to or close to the

raw method. Medium means the operation time of the implemented solution is higher than the raw method but is acceptable for practical implementation. Low means the operation time is much higher than the raw method and that the solution needs further improvement for practical implementation.

GPS spoofing attack has been widely studied in secure positioning research. Focusing on GPS spoofing attacks in aviation, Kai et al. [35] implemented effective attack detection and tracing solutions by leveraging crowdsourced air traffic monitoring sensor networks. The proposed solutions showed high efficiency with attack detection in 2 s and attacker tracing within 15 min. For a GPS spoofing attack, Manesh et al. [56] achieved attack detection using a model trained with features like pseudo-range, Doppler shift and signal-to-noise ratio (SNR). Wang et al. [95] presented an attack-reduction solution by using GPS signal reconstruction technology. Both their solutions showed high efficiency with only seconds of latency. Different from the above work, Singh et al. [87] designed the first attack-reduction solution against distance enlargement and reduction attacks. In addition, the designed system could be implemented directly on top of existing 5G-NR transceivers. However, the solution only achieved medium efficiency because of the communication delay between transceiver and receiver. In comparison, the current dissertation based on Publication VI focuses on jamming and collusion attacks of the 5G positioning system; it is the first work that supports both attack reduction, detection, classification and tracking. The solutions have been implemented with clustering methods and neural network models, hence preserving high-efficiency performance in terms of positioning latency.

Privacy in 5G Positioning

Table 2.2 compares the related work on preventing information leakage in 5G positioning. For a comparison, the related work has been evaluated by looking at privacy, method, third party, and efficiency:

- Based on its role, the participants of 5G positioning can be divided into target points and reference points. The target point is the one requesting positioning service, and the reference points are the ones assisting with the positioning service. The Privacy of target points and reference points refers to the confidentiality of their location information during positioning.
- The method describes the applied technology to achieve privacy protection.
- The third party is marked as check when a third party is involved in the solution, such as a trusted key distributor. The requirement of a

Table 2.2. Related work comparison w.r.t. 5G positioning privacy protection

Ref.	Privacy		Method	Third party	Performance
	Target	Reference			
[75]	✓		k-anonymity	✓	Medium
[47]	✓	✓	Paillier encryption		Low
[105]	✓		Garbled circuit		Low
[33]	✓	✓	Paillier encryption		Low
[85]	✓		Homomorphic encryption		Low
[36]	✓	✓	Paillier encryption		Low
[72]		✓	One-pass authentication		High
[89]	✓	✓	Feature extraction		High
PI	✓	✓	Matrix multiplication and concatenation		High

third party increases the difficulty of the practical implementation of the solution.

- The performance is defined as discussed in Table 2.1.

Sazdar et al. [75] proposed a scheme based on k-anonymity for indoor positioning protection. The query from a target point was first mapped into a Bloom filter vector and further obfuscated with another $k - 1$ random vectors. Thus, the location of the target point could be hidden with random values. Although privacy can be protected with k-anonymity, it requires a trusted third party for obfuscation. Also, it decreases the positioning performance because the computation and communication cost occur over random values. Also for indoor positioning, Li et al. [47] implemented a privacy-preserving solution for fingerprint-based localisation by leveraging the Paillier cryptosystem. In this way, the positioning computation can be processed in the ciphertext, and the privacy of both target and reference points can be protected. Likewise, based on another cryptographic tool (garbled circuit), Yang and Järvinen [105] proposed a solution for indoor positioning, and Hussain and Koushanfar [33] proposed a solution for

Table 2.3. Related work summary w.r.t verifiable positioning

Ref.	Contribution	Method	Performance
[25]	Model verification	Interactive proof protocol	Low
[110]	Computation verification	Homomorphic encryption and polynomial factorisation	Low
[15]	Position verification	Distance comparison	High
[79]	Position verification	Moving to claimed locations	High
PII	Model verification	Backdoor technique	High

vehicle positioning. Shu et al. [85] also implemented a privacy protected location computation with matrix multiplication and homomorphic encryption. Similar work based on Paillier encryption can also be found in [36]. Even though these cryptographic-based solutions were independent of the third party and provided a high level of privacy, their performance was low because of the heavy cost generated by cryptography operations. Different from the above work, Pei et. al. [72] enhanced the privacy protection of reference points in 3D positioning by using a one-pass authentication key exchange protocol. Even with a little extra cost, the solution was still able to maintain high performance. A privacy preservation scheme for image-based localisation was presented by Speciale et al. [89] to avoid confidential information leakage from 3D images. The privacy of the image can be protected by transferring it to a 3D line representation with effective feature extraction strategies. The scheme protects privacy of both the target and reference points, hence providing a high level of performance. Based on Publication I, the current dissertation provides two protocols (Pub-pos and Pri-pos) that satisfy the privacy protection of 5G positioning. Pub-pos focuses on the privacy protection of the target point, while Pri-pos protects the privacy of both the target and reference point. Both Pub-pos and Pri-pos solve the above problems without needing a third party. In addition, Pub-pos and Pri-pos provide a high level of performance both theoretically and experimentally compared with existing solutions.

2.2.2 Verifiable Positioning

The related work of verifiable positioning is summarised in Table 2.3 regarding the contribution, method and performance:

- The contribution describes the target problem solved in the related

work.

- The method describes the applied technology to achieve the goal.
- The performance is defined as discussed in Table 2.1.

Since its first appearance in 1991 [9], verification has been widely studied in the field of outsourced cloud computing, and extensive solutions have been proposed to address verification problems. Based on an interactive proof protocol, Ghodsi et al. [25] proposed SafetyNets, a framework enabling users to check the correctness of outsourced deep learning models by challenging a prover (normally the cloud) to solve some computational hard challenges. However, its performance was relatively low, and the complexity to solve the hard challenges was often orders of magnitude more than model prediction computations [93]. On the other hand, Yu et al. [110] proposed a verifiable outsourced computation scheme over encrypted data with the help of fully homomorphic encryption and polynomial factorisation. However, the performance was low because of the large amount of overhead introduced by cryptography operations. Capkun and Hubaux [15] and Schfer et al. [79] focused on providing solutions for users to verify their position information. In [15], the user verified its position information by comparing the computed distance (between their device and a set of known reference points) with their real distance (measured by hand). The position information was accepted only when they were the same. However, this solution greatly depended on the existence of reference points. To minimise the dependence on the reference points, the user in [79] had to verify its position by moving to certain locations. As for performance, they preserved high efficiency compared with other verification solutions. However, these two solutions were not feasible for an integrity check of the outsourced positioning model at the edge point. The current literature has not yet studied this issue and provided a feasible solution. The current dissertation is the first work to achieve this goal. Different from traditional methods using secure processors or trusted platforms, the solution based on Publication II adopts a simple but effective backdoor technique to solve the problem. Also, the results showed that the proposed solution presents a high level of efficiency in terms of performance.

2.2.3 Privacy Protection in D2D Location Verification

A related work summary of privacy protection in D2D location verification is given in Table 2.4. The work is evaluated under the criteria of privacy, method, third party and performance:

- Privacy protection in D2D location verification includes the protection of the location and measured distance because they are both sensitive information that threatens the privacy of participants.

Table 2.4. Related work w.r.t privacy protection in D2D location verification

Ref.	Privacy		Method	Third party	Performance
	Location	Distance			
[102]	✓		Access Control		High
[64]	✓		<i>K</i> -anonymity	✓	High
[50]	✓		Dummy		High
[65]	✓		Dummy		High
[24]	✓		Private information retrieval		Low
[63]	✓		Decision Diffie-Hellman		Low
PIII	✓	✓	Order-preserving encryption		Medium

- The method describes the applied technology to achieve privacy protection. Here, it is classified as access control, obfuscation and cryptography.
- Third party is marked when a third party is involved.
- The performance is defined as discussed in Table 2.1.

Xu et al. [102] achieved location protection using strict access control. Only users who met predefined policies or trust requirements had access to the location information. The solution presented a high level of performance, but it was vulnerable to attacks from compromised participants. Niu et al. [64] proposed a method to protect a user's location with *K*-anonymity. However, the solution required a trusted local server for random value generation. In contrast, the dummy-based solution proposed by [50] and [65] shared the same idea as *K*-anonymity but without the requirement of a third party. Both *K*-anonymity and dummy-based solutions showed high levels of performance. However, they were vulnerable to spatio-temporal correlation attacks when users submit consecutive requests. Ghinita et al. [24] and Narayanan et al. [63] focused on cryptography-based solutions. Ghinita et al. [24] implemented a location-protected nearest neighbour search based on private information retrieval (PIR), and Narayanan et al. [63] implemented a location protected equality testing protocol based on the Decision Diffie-Hellman (DDH) problem. However, their performance was quite low because of the high compu-

Table 2.5. Related work summary w.r.t privacy-preserving LBS

Ref.	Privacy		Method	Third	Service	Perfor- mance
	UE	LBSP				
[5]	✓		A (k-anonymity)		kNN, range query	High
[16]	✓	✓	A (dummy)	✓	kNN, range query + key-word	High
[46]	✓	✓	A (dummy)	✓	kNN, range query + key-word	High
[49]	✓		C (homomorphic)		Crowdsourcing tasks assignment	Low
[111]	✓	✓	C (symmetric)		Crowdsourcing tasks assignment	Medium
[96]	✓	✓	C (SMC)		Recommendation	Low
[116]	✓	✓	C (SGX)		kNN, range query + key-word	Medium
[90]	✓		N (DP)		data releasing	High
[41]	✓		N (LDP)		Fuzzy kNN	High
[7]	✓		N (LDP)		Fuzzy kNN, range	High
PIV	✓	✓	C (OT and CP-ABE)		kNN + key-word	Medium
PV	✓	✓	N (LDP)		Fuzzy kNN + keyword	High

tation overhead of PIR and DDH. The current dissertation aims to protect both location and distance information in D2D location verification. As presented in Publication III, the solution introduces an innovative coordinate-based verification method and privacy protection solution based on order-preserving encryption. Compared with traditional solutions, this solution achieves the best balance between privacy and efficiency. Also, no third party is needed.

2.2.4 Privacy-Preserving Location-Based Service

This subsection reviews the related work of privacy protection in LBS. A summary of the related work is provided in Table 2.5. For comparison, the work is evaluated with the following criteria: privacy, method, third party, query, keyword and performance.

- Privacy protection in LBS includes the protection of both the UE and LBSP. The UE requires their query data, especially position information, to be protected because the disclosure of this information could put them at risk [48,84]. The LBSP wants to protect its collected database from any unauthorised UEs. A mutual protected solution is expected [51].
- The method describes the applied technology to achieve the goal. A, C and N represent anonymity, cryptography and noise injection, respectively.
- Third party is marked when the solution requires a third party.
- Query implies the supported LBS queries, which can be NN query, range query, kNN query and so forth [104].
- The keyword is checked when text-based keyword searching is supported. Keyword examples can be 'cafe, supermarket'.
- The performance is defined as discussed above in Table 2.1.

Anonymity-based (A) solutions are represented by k-anonymity and dummy, which use a fake query to disguise a real query. Based on a dummy mechanism, Alharthi et al. [5] proposed an anonymous communication technique for crowd-workers in which their location privacy was protected. Anonymous communication can support kNN and range queries with a high level of performance; however, the database of the LBSP was not protected. To achieve the protection of the LBSP, Chen et al. [16] and Kuang et al. [46] employed a semi-trusted third party to generate redundant data to protect query privacy and filter out redundant responses. However, the introduction of a third party was not recommended for the practical implementation of the solution. Cryptography-based solution (C) has also been widely used in privacy protection. Based on homomorphic encryption, Liu et al. [49] implemented a secure spatial crowdsourcing framework protecting the exact location of tasks and workers while assigning crowdsourcing tasks. However, it could not support the privacy of the LBSP, and the performance was low because of homomorphic encryption. To protect LBSP privacy in crowdsourcing task assignments, Yuan et al. [111] used a grid-based method for protecting location privacy and applied symmetric encryption for preserving the privacy of task content.

Their experimental results showed a medium level of performance. Based on secure multi party computation (SMC), Wang et al. [96] implemented a POI recommendation system that considered data privacy for both the UE and LBSP. However, the heavy computations resulted in low levels of performance. Yan et al. [116] was the first to work with software guard extensions (SGX) for LBS privacy protection. SGX supports all kinds of LBS queries while protecting the privacy of both UE and LBSP. In addition, the solution achieved a medium level of performance. However, their practical implementation was still limited because of the dependence on SGX.

Noise injection (N) protects privacy by adding random noise into original data while preserving data utility. A differential privacy (DP)-based data releasing framework was proposed by To et al. [90] to protect worker locations in spatial crowdsourcing. To prevent the leakage of visiting frequency, Kim et al. [41] integrated local differential privacy (LDP) into a successive POI prediction model. Andrés et al. [7] also proposed geoindistinguishability based on LDP as a solution for location protection in general cases. Although these solutions were shown to be effective for location protection, they only supported fuzzy queries because of the influence of random noise and also were unable to handle privacy protection of keyword information. Nevertheless, this type of solution can greatly diminish computation and communication overhead compared with other methods. Based on Publication IV and Publication V, the current dissertation presents the privacy protection of the representative kNN query while providing mutual privacy to the LBSP and UEs. In comparison, the proposed solutions are independent of the third party and all support keyword search. Most importantly, the efficiency of the proposed protocols is above average compared with the others. Technically, Publication IV implements a kNN query based on cryptography (OT and CP-ABE) and Publication V designs a fuzzy kNN query with a noise injection LDP method.

2.3 Summary

This chapter gave a brief introduction to the background technologies (OPE and LDP) and the evaluation criteria. It also investigated the related work corresponding to our research problems.

3. Security and Privacy Protection in 5G Positioning

This chapter aims to identify the vulnerabilities in the existing 5G positioning system, proposing solutions to defend against these potential attacks.

3.1 Problem Statement

According to Fig. 1.2, the 5G positioning system involves three main parties: the UE with a 5G connection, a variety of ANs (also known as reference points) and a FC deployed by the network. The UE transmits periodical uplink pilot signals to ANs. The AN extracts the directional and temporal parameters, that is, ToA and DoA of the UE. These parameters are delivered to an FC that carries out the computation of the UE position. Note that the FC is a networked device that can carry out computation, has storage and communication connection. The AN can be a roadside unit, a base station or even an outsourced cloud service provider. Eventually, the position information is sent back to the UE through downlink beams or by the FC directly.

Based on this assumption, we consider two main security attacks.

- *Radio jamming attack*: This is also known as man-in-the-middle attack. We assume the adversary can intercept the positioning signalling path between the UE and AN; thereby, an attacker can alter the signalling by transmitting energy to disrupt reliable data communications, which is normally carried out through a jammer. It is notable that by resisting this attack, the positioning system will become more robust to unintentional interference.
- *Collusion attack*: We assume an adversary can hack or even set up fake ANs in 5G networks and send wrong or erroneous signal measurements to deteriorate the positioning system. By detecting a collusion attack, we can effectively eliminate the presence of malicious nodes in the system.

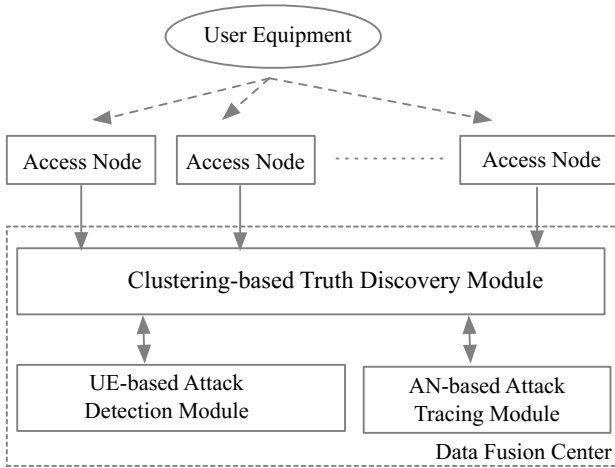


Figure 3.1. Secure positioning with truth discovery, attack detection and tracing

Besides these security attacks, we also consider the risk of privacy disclosure. The threat in positioning resides in three aspects:

- *The estimated coordinates of the UE:* This results from the sensitivity of the location and the possibility of inferring sensitive information related to the location owner. For example, a user’s home address could be inferred if their location is in a residential area, or their health status could be inferred if their location is at a hospital for some specific disease treatment.
- *The coordinates of the ANs:* This should be kept secret to prevent the potential information leakage of access nodes. However, an exception also exists when the ANs are public infrastructures with no privacy required, such as base stations and roadside units in VANET.
- *The distance between UE and ANs:* This is treated as sensitive for two reasons. First, it risks the privacy of the UE because adversarial attackers can use it to locate the target point when combined with public ANs. Second, it risks the privacy of the AN because a malicious target point could use it to locate a private AN based on historical interaction data. Thus, distance information is sensitive and should be kept secret.

3.2 Secure Positioning with Truth Discovery, Attack Detection and Tracing

Fig. 3.1 shows the scheme overview. In network-based positioning, the whole scheme process is operated by the FC, and it contains three modules:

truth discovery, attack detection and tracing. The truth discovery module is initiated once all the required parameters are collected from the ANs. It aims to cluster collected parameters and calculate their true value by removing outliers. Based on the clustering result, attack detection and tracing can be further performed. The detailed design of each module is described below and the notations of this section are summarized in Table 3.1.

Table 3.1. Notations

Symbol	Notation
i, j	The identifications of the AN and UE
(X_i, Y_i)	The coordinates of AN_i
$(ToA_{i,j}, DoA_{i,j})$	Time of arrival and direction of arrival between the UE and AN
ID	The identification of signal data
\mathcal{D}	The location set for UE
$C = \{C_1, C_2, \dots, C_K\}$	The cluster result
ε	The threshold of distance in DBSCAN
min	The minimum items in DBSCAN clusters
P	The central point of cluster
RQ	The output of range query
U	The training dataset of attack detection
(x, y)	The estimated position of UE
\mathcal{L}	The label generated by truth discovery
t	The number of 'Noise' data
n	The number of ANs in positioning
(x_i, y_i)	The real position of UE
(\bar{x}, \bar{y})	The approximate position estimated by FC
A	The training dataset of attack tracing
a	The record of dataset A
T	Time period for signal collection
\mathcal{R}	The detection results of all ANs
M	The number of sampling data

3.2.1 Truth Discovery

The main idea of truth discovery is to label the data collected from ANs using a clustering algorithm [14] and then calculating the position based

only on non-noise data. The process is composed of three phases:

Position Parameter Collection

To obtain the positioning service, the UE periodically transmits uplink pilot signals to surrounding ANs. Once receiving the pilot signals, ANs extract position parameters such as DoA and ToA from them and upload the estimated parameters to the FC through a secure communication channel. The transmitted message between an AN and the FC is represented as follows:

$$\{(X_i, Y_i), (ToA_{i,j}, DoA_{i,j}, AN_i, UE_j, ID)\},$$

where i and j are the identifications of the AN and UE, respectively; (X_i, Y_i) denotes the coordinates of AN_i ; $(ToA_{i,j}, DoA_{i,j})$ denote the extracted parameters of UE_j from AN_i , and ID is the signal identification. For UE_j positioning, we suppose there are multiple ANs included for its position calculation and that the ANs' signals related to UE_j are labelled with same ID generated by the AN.

True Value Detection

Based on the received parameters from AN_i , the FC estimates the position of UE_j using the following formula:

$$\begin{cases} x_j = (ToA_{i,j} * c) * \cos(DoA_{i,j}) + X_i \\ y_j = (ToA_{i,j} * c) * \sin(DoA_{i,j}) + Y_i \end{cases} \quad (3.1)$$

Thus, for every UE_j , there will be a location set

$$\mathcal{D} = \{(x_{j,1}, y_{j,1}), (x_{j,2}, y_{j,2}), \dots, (x_{j,n}, y_{j,n})\},$$

where n is the number of ANs involved in the positioning process.

However, the above locations may be inaccurate because of the existence of various attacks. To eliminate their influence, we use a clustering algorithm to detect outliers. The algorithm is presented in Alg. 1 and is designed based on Density-based Spatial Clustering of Applications with Noise (DBSCAN) [77] for sound performance and adaptability in clustering. For the distance measurement in clustering, we adopt the euclidean distance [21], which has been widely used in distance measurement in highly dimensional scenarios.

In the algorithm, we first initialise the scan radius ε and minimum items min . As a rule of thumb, min can be derived from the number of dimensions in the data set, as $min \geq dimension + 1$. The value of ε can be chosen by using k-distance graph, where $k = min - 1$. In general, a large value of min and a small of value of ε produce better clustering result. Meanwhile, all the locations are labelled as '*Initial*' when they are not processed. Then, we iterate over every estimated point in \mathcal{D} . For each point $P \neq 'Initial'$, we calculate the number of points included in a circle, with

Algorithm 1: Clustering-Based True Value Detection

Input: $\mathcal{D} = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$
Output: $C = \{C_1, C_2, \dots, C_K\}$

- 1: initial ε, min, C
- 2: **for** each point P in \mathcal{D} **do**
- 3: **if** $P \neq 'Initial'$ **then**
- 4: $continue$
- 5: **end if**
- 6: Neighbors $RQ = \text{RangeQuery}(\mathcal{D}, P, \varepsilon)$
- 7: **if** $|N| \leq min$ **then**
- 8: $label(P) = 'Noise'$
- 9: $continue$
- 10: **end if**
- 11: $C = C + C_{new}$
- 12: $C_{new} = C_{new} + P$
- 13: $S = RQ \setminus P$
- 14: **for** each point Q in S **do**
- 15: **if** $Q = 'Noise'$ **then**
- 16: $C_i = C_i + Q$
- 17: **end if**
- 18: **if** $Q \neq 'Noise'$ **then**
- 19: $continue$
- 20: **end if**
- 21: $C_{new} = C_{new} + Q$
- 22: Neighbors $RQ = \text{RangeQuery}(\mathcal{D}, P, \varepsilon)$
- 23: **if** $|N| \leq min$ **then**
- 24: $label(P) = 'Noise'$
- 25: $continue$
- 26: **end if**
- 27: **end for**
- 28: **end for**

P as the center and ε as its radius, here by calling function `RangeQuery` (in Alg. 2), which returns a point set noted as RQ . If the number of points in RQ is smaller than min , point P is labelled ' $Noise$ '; otherwise, a new cluster C_{new} is created with P as the included item. Then, another round of `RangeQuery` is conducted on all the neighbours of P , which are stored in RQ . The process stops when all the points in \mathcal{D} are processed and the result of the algorithm is returned as the K clusters in C .

Position Calculation

Because we suppose the amount of affected signals caused by interference or corrupted ANs is always less than half in each positioning ($\leq n/2$), the

Algorithm 2: RangeQuery

Input: $\mathcal{D}, P, \varepsilon$ **Output:** RQ

```

1: Neighbors  $RQ = \text{null}$ 
2: for each point  $Q$  in  $\mathcal{D}$  do
3:   if  $\text{Euclidean}(Q, P) \leq \varepsilon$  then
4:      $RQ = RQ \cup \{P\}$ 
5:   end if
6: end for
7: return  $RQ$ 

```

cluster in C with the most items (more than 50%) is the one containing clean data. The real location of UE_j can be calculated by averaging the points from this cluster and returning it to UE_j . The rest of the points not included in the calculation are labelled as noise data and will be used as an input for attack detection and tracing.

3.2.2 Attack Detection

Jamming attack intercepts the communication channel between the UE and AN by injecting noise signals through a jammer. It is especially dangerous when the attacker knows the UE's movement, because the attacker can follow the UE and keep attacking the UE. The attack detection module is designed for finding attacked UEs so that the attacked UEs can be alerted as soon as possible.

Our solution is to train a machine learning model that can perform attack detection. The training dataset U is historical positioning data collected by the FC. For simplicity, we denote the record of U as $u = \{(x_1, y_1, l_1), (x_2, y_2, l_2), \dots, (x_n, y_n, l_n)\}$, where (x, y) is the position information of the UE as estimated by the n th ANs and \mathcal{L} is the label generated by truth discovery. Based on the training dataset, feature extraction and model training are conducted.

Feature Extraction

Feature extraction is an essential step for generating formatted, nonredundant and informative data to facilitate model training. Four distinctive features are applied for attack detection:

(a) Abnormal Positioning Data Ratio (APDR)

This refers to the percentage of 'Noise' signals in each UE's positioning. For each record, it is calculated as $APDR = t / n$, where t is the number of 'Noise' data and n the total number of ANs participating in positioning. When the UE is under attack, APDR increases with increased 'Noise' data related to the attacked UE.

(b) Abnormal Positioning Status (APS)

This is set as 1 when noise data are over 20% in each positioning; otherwise, it is 0. Normally, APS is activated as 1 when the communication channel between the UE and AN is under attack. This feature is selected to complement the APDR. When an attack only influences one or two points of data and the APDR is relatively small, the attack may be ignored. The APS is designed to capture this situation.

(c) Positioning Error Mean (PEM)

This refers to the mean error between the measured positions and the FC estimated approximate position from truth discovery in each record. The computation of PEM is as follows:

$$PEM = \frac{\sum_{j=1}^n \|(x_i, y_i) - (\bar{x}, \bar{y})\|}{n},$$

where (x_i, y_i) is the measured position and (\bar{x}, \bar{y}) is the FC estimated approximate position. It remains low when the UE is in a normal state and increases when the UE is attacked.

(d) Positioning Error Variance (PEV)

This refers to the mean variance between measured and estimated positions. The PEV is a necessary feature as it reflects the deviation of position data when the PEM is relatively small. However, PEM detection is invalid when the attackers try to control the PEM at a low level by occasionally turning off the attack. When the PEV is included in the feature set, we can detect such an attack.

Model Training

We design the neural network model with four inputs and one output. The training data are composed of the above specified features and labelled as 0 or 1 according to UE status. Here, 0 indicates a normal state, while 1 indicates that the UE is under attack. Normalisation is applied to each feature so that the difference between features can be balanced; thus, they contribute similarly to UE status prediction. After training, the model can be used directly. When in use, we collect the observation data of the UE and generate the same features from these data. With these features as the input, the model outputs 0 or 1, here corresponding to a normal or attacked status of the UE.

3.2.3 Attack Tracing

Another crucial implementation is attack tracing, which aims to detect corrupted ANs. It is essential because it helps locate the sources of attacks. Likewise, the problem can be solved by applying machine learning. The training dataset A contains the historical data of ANs. We denote each record as $a = \{(x_1, y_1, l_1, t_1), (x_2, y_2, l_2, t_2), \dots, (x_n, y_n, l_n, T)\}$, which includes the positioning data uploaded by an AN in a time period T .

Feature Extraction

Based on dataset A , we extract and train the model of attack tracing according to the following three typical features:

(a) Abnormal Upload Ratio (AUR)

This refers to the percentage of 'Noise' data uploaded by an AN. When an AN is attacked or corrupted, its data are dirty, here with a high probability. The value of the AUR is relatively small when an AN is in a normal state but increases when an AN is under attack or malfunctions.

(b) AN Positioning Error Mean (APEM)

The APEM refers to the error mean between the estimated and approximate position. It is calculated as follows:

$$APEM = \frac{\sum_{i=1}^M \|(x_i, y_i) - (\bar{x}, \bar{y})\|}{M},$$

where (x_i, y_i) is the position measured by the AN, (\bar{x}, \bar{y}) is the approximate position of the UE estimated by FC, and M is the number of uploaded signals from the AN in the time period T . The APEM is chosen as a typical feature because attacks on an AN result in large deviations between the position measured by the AN and the approximate position, which causes the APEM to increase.

(c) AN Positioning Error Variance (APEV)

This refers to the error variance between the estimated and approximate positions. Regarding malicious ANs, they can evade the detection based on the APEM by using on-and-off attacks, which means that the APEM can be controlled within an acceptable range. When the APEV is included in the feature set, it overcomes the shortcoming of the APEM for detecting the on-and-off attack.

Model Training

We design a neural network with three inputs and one output. The training data are also labelled as 0 or 1 according to the status of the AN, where 0 means that the AN is normal and 1 means it is attacked. When in use, for each AN involved in the positioning of the UE, the FC extracts the features from its uploaded data in period T and inputs them into the trained model. \mathcal{R} is obtained as the output from n ANs. AN_i is functioning normally when \mathcal{R}_i is 0 and under attack if \mathcal{R}_i is 1. In this way, we can locate all the ANs under attack and fix them.

3.3 Efficient Privacy Protection Protocols for 5G-Enabled Positioning

To mitigate the positioning privacy risk in emerging technologies like D2D, V2X, crowdsourcing and even unmanned aerial vehicles, two protocols, named Pub-pos and Pri-pos, with different privacy constraints are pro-

posed. Pub-pos is a privacy-preserving positioning protocol that assumes the public availability of ANs' locations. It aims to prevent the privacy leakage of measured distance and compute the location of ANs. Pri-pos is designed for a scenario where the locations of ANs are also sensitive and need to be protected. The notations of this section are summarized in Table 3.2.

Table 3.2. Notations

Symbol	Notation
U	The end-user with 5G access
R_i	The reference point i
S	The computation server
$\mathcal{X} = (x_i, y_i, z_i)$	The coordinates
d_i	The distance between UE and AN i
RA_i	The random matrix for coordinate information of R_i
RB_i	The random matrix for distance information of R_i
K^A	The random key matrix for coordinate information of R_i
K^B	The random key matrix for distance information of R_i
l, k	The dimension and bitlength of random matrix K^B
s	The dimension of random matrix K^A
B_1, B_2, B_3, B_4	The matrix of distance from reference points
A_1, A_2, A_3, A_4	The matrix of coordinate from reference points
C, D, E, F, G	The middle value computed by R and S
d	The decryption key from K^B
H	The decryption key from K^A

3.3.1 Pub-pos: Privacy-Preserving Positioning Based on Public Access Nodes

Pub-pos assumes that the coordinates of each AN are public. They obtain their coordinates from GPS and publicise them for assisting in positioning services. However, the distance information held by these ANs is sensitive and must be kept secret from the computation server. The computation server is responsible for collecting the positioning-related data from each AN and sending its computation result to the target UE. Notably, such a localisation model is popularly used in outdoor positioning based on base stations and VANET positioning based on roadside units.

Algorithm 3: Privacy-Preserving Positioning Based on Public Access Nodes

Result: U with coordinates \mathcal{X}

Private Inputs: R_i with distance d_i

Public Inputs: R_i with coordinates (x_i, y_i, z_i) ;

Step1 (Target UE U):

1.1: generate four $3 \times l$ matrices RB_1, RB_2, RB_3, RB_4 with k-bits random value and $RB_1 = RB_2 + RB_3 + RB_4$;

1.2: generate a $(l + 1) \times 1$ matrix K^B with k-bits random value ;

1.3: distribute $(RB_1, K^B), (RB_2, K^B), (RB_3, K^B), (RB_4, K^B)$ to reference points R_1, R_2, R_3, R_4 randomly;

Step2 (ANs R_1, R_2, R_3, R_4):

2.1: each accepts key pair (RB_i, K^B) respectively;

2.2: each compose a matrix with its distance d as

$$B_1 = \begin{bmatrix} d_1^2 \\ d_1^2 \\ d_1^2 \end{bmatrix}, B_2 = \begin{bmatrix} d_2^2 \\ 0 \\ 0 \end{bmatrix}, B_3 = \begin{bmatrix} 0 \\ d_3^2 \\ 0 \end{bmatrix}, B_4 = \begin{bmatrix} 0 \\ 0 \\ d_4^2 \end{bmatrix}$$

2.3: each merge B and RB by row concatenation, as $D = \begin{bmatrix} B & RB \end{bmatrix}$;

2.4: each encrypt D with K^B as $E = DK^B$;

2.5: each sends it E denoted as E_1, E_2, E_3, E_4 to the computation server S ;

Step3 (Computation Server S):

3.1: accept E_1, E_2, E_3, E_4 ;

3.2: compose A and B_5 based on public coordinates as

$$A = 2 \begin{bmatrix} (x_1 - x_2) & (y_1 - y_2) & (z_1 - z_2) \\ (x_1 - x_3) & (y_1 - y_3) & (z_1 - z_3) \\ (x_1 - x_4) & (y_1 - y_4) & (z_1 - z_4) \end{bmatrix}$$

$$B_5 = \begin{bmatrix} -x_2^2 - y_2^2 - z_2^2 + x_1^2 + y_1^2 + z_1^2 \\ -x_3^2 - y_3^2 - z_3^2 + x_1^2 + y_1^2 + z_1^2 \\ -x_4^2 - y_4^2 - z_4^2 + x_1^2 + y_1^2 + z_1^2 \end{bmatrix}$$

3.3: compute $C = A^{-1}(-E_1 + E_2 + E_3 + E_4), F = A^{-1}B_5$;

3.4: send C and F to U ;

Step4 (Target U):

4.1: accept C and F ;

4.2: reveal the location as $\mathcal{X} = C\mathbf{d}^{-1} + F$ where $\mathbf{d} = K_1^B$;

Protocol Design

The protocol to achieve the above requirements is presented in Algorithm 3. The target UE U starts the protocol by generating four $3 \times l$ matrices

RB_1, RB_2, RB_3 and RB_4 and a $l + 1 \times 1$ matrix K^B . It is required that $1 \leq l$ so that random matrix RB is not empty. Also, $13l + 1 \leq 2^k$ so that there are enough random values for selection. It is also required that $RB_1 = RB_2 + RB_3 + RB_4$ and that all random values in the matrix are no longer than k -bits. Herein, the selection of random values distribution is flexible. We generate random values following a uniform distribution, but other distributions like Gaussian or Laplace can also be applied because the privacy of our protocol can be preserved without depending on the distribution of the random values. U then distributes (RB_1, K^B) , (RB_2, K^B) , (RB_3, K^B) and (RB_4, K^B) randomly to four ANs (R_1, R_2, R_3, R_4) through secure channels. The AN R_i accepts RB_i and K^B . At the same time, it composes its distance information d_i into a 3×1 matrix B_i , as shown in step 2.2 of Algorithm 3. To randomise the distance matrix B_i , R_i first merges it with RB_i by row concatenation and further encrypts the merged matrix with the random matrix K^B . The encrypted matrix denoted as E_i can then be sent to the computation server S . At S , A^{-1} and B_5 are first initialised according to the public knowledge of $\{(x_1, y_1, z_1), (x_2, y_2, z_2), (x_3, y_3, z_3), (x_4, y_4, z_4)\}$, as shown in step 3.2 of Algorithm 3. Once accepting the encrypted matrices from all reference points, S integrates the obtained matrix B_1, B_2, B_3, B_4 with A^{-1} as $C = A^{-1}(-B_1 + B_2 + B_3 + B_4)$. Meanwhile, it computes $F = A^{-1}B_5$. Once completed, both C and F are sent back to U . Based on C and F , U reveals its location with $\mathcal{X} = C\mathbf{d}^{-1} + F$, where $\mathbf{d} = K_1^B$, as shown in step 3.2 of Algorithm 3.

Correctness Analysis

The correctness analysis can be derived with the following equation by substituting the formulas:

$$\begin{aligned} \mathcal{X} &= C\mathbf{d}^{-1} + F \\ &= A^{-1}(-E_1 + E_2 + E_3 + E_4)\mathbf{d}^{-1} + A^{-1}B_5 \\ &= A^{-1}(-D_1 + D_2 + D_3 + D_4)K^B\mathbf{d}^{-1} + A^{-1}B_5 \end{aligned} \quad (3.2)$$

First, we prove that

$$(-D_1 + D_2 + D_3 + D_4)K^B\mathbf{d}^{-1} = -B_1 + B_2 + B_3 + B_4.$$

Because $D_i = \begin{bmatrix} B_i & RB_i \end{bmatrix}$ which is

$$\begin{bmatrix} d_i^2/0 & r_{11}^i & r_{12}^i & \cdots & r_{1l}^i \\ d_i^2/0 & r_{21}^i & r_{22}^i & \cdots & r_{2l}^i \\ d_i^2/0 & r_{31}^i & r_{32}^i & \cdots & r_{3l}^i \end{bmatrix}$$

and $RB_1 = RB_2 + RB_3 + RB_4$, it is easy to derive that $-D_1 + D_2 + D_3 + D_4$

is

$$\begin{bmatrix} d_2^2 - d_1^2 & 0 & 0 & \cdots & 0 \\ d_3^2 - d_1^2 & 0 & 0 & \cdots & 0 \\ d_4^2 - d_1^2 & 0 & 0 & \cdots & 0 \end{bmatrix}$$

Let $K^B = [k_1 \ k_2 \ \cdots \ k_l \ k_{l+1}]^T$, we have

$$(-D_1 + D_2 + D_3 + D_4)K^B = \begin{bmatrix} (d_2^2 - d_1^2)k_1 \\ (d_3^2 - d_1^2)k_1 \\ (d_4^2 - d_1^2)k_1 \end{bmatrix}$$

Also because $\mathbf{d} = K_1^B = k_1$, and $(-D_1 + D_2 + D_3 + D_4)K^B \mathbf{d}^{-1}$ is

$$\begin{bmatrix} d_2^2 - d_1^2 \\ d_3^2 - d_1^2 \\ d_4^2 - d_1^2 \end{bmatrix}$$

Thus, $(-D_1 + D_2 + D_3 + D_4)K^B \mathbf{d}^{-1} = -B_1 + B_2 + B_3 + B_4$ is proved. Based on this proof, the following equation can be derived from Equation 3.2:

$$\begin{aligned} \mathcal{X} &= A^{-1}(-B_1 + B_2 + B_3 + B_4) + A^{-1}B_5 \\ &= A^{-1}(-B_1 + B_2 + B_3 + B_4 + B_5) \end{aligned}$$

According to the definition of B_1, B_2, B_3, B_4 and B_5 , we have $B = -B_1 + B_2 + B_3 + B_4 + B_5$ as

$$\begin{bmatrix} d_2^2 - x_2^2 - y_2^2 - z_2^2 - d_1^2 + x_1^2 + y_1^2 + z_1^2 \\ d_3^2 - x_3^2 - y_3^2 - z_3^2 - d_1^2 + x_1^2 + y_1^2 + z_1^2 \\ d_4^2 - x_4^2 - y_4^2 - z_4^2 - d_1^2 + x_1^2 + y_1^2 + z_1^2 \end{bmatrix}$$

Thus,

$$\mathcal{X} = A^{-1}B.$$

Thus, the correctness of the protocol can be proved.

Privacy Analysis

This part proves that the protocol presented in Algorithm 3 can preserve the privacy of (1) the distance d_i of each access node and (2) estimated coordinates \mathcal{X} of the target UE. The theorem and proof are given below.

Theorem 3. *For all access nodes, the distance d_i to the target UE U is protected from computation server S .*

For the access node R_i , it accepts random matrices K^B and RB from U , using them to encrypt its distance information d_i , as shown in steps 2.3 and 2.4 of Algorithm 3. The computed matrix E_i is then sent to the computation server S . We need to prove that S is unable to infer any information about d_i from either E_i or its combination. To have a clear picture, we present D_i and K^B as follows:

$$D_i = \begin{bmatrix} d_1^i & r_{11}^i & r_{12}^i & \cdots & r_{1l}^i \\ d_2^i & r_{21}^i & r_{22}^i & \cdots & r_{2l}^i \\ d_3^i & r_{31}^i & r_{32}^i & \cdots & r_{3l}^i \end{bmatrix}$$

$$K^B = \begin{bmatrix} k_1 & k_2 & \cdots & k_l & k_{l+1} \end{bmatrix}^T$$

where d^i is the distance information and r and k are bit lengths of random values. Based on D_i and K^B , we express E_i as

$$E_i = \begin{bmatrix} d_1^2 k_1 / 0 + r_{11}^i k_2 + r_{12}^i k_3 + \cdots + r_{1l}^i k_{l+1} \\ d_2^2 k_1 / 0 + r_{21}^i k_2 + r_{22}^i k_3 + \cdots + r_{2l}^i k_{l+1} \\ d_3^2 k_1 / 0 + r_{31}^i k_2 + r_{32}^i k_3 + \cdots + r_{3l}^i k_{l+1} \end{bmatrix}$$

Observing E_i , it contains $4l + 2$ variables. It is impossible to reverse either d , r or k through matrix decomposition because there are insufficient equations compared with variables. Thus, it is impossible for S to infer information about R_i from a single E_i . More information comes from the fact that $RB_1 = RB_2 + RB_3 + RB_4$. This can be used to remove the random r with

$$-E_1 + E_2 + E_3 + E_4 = \begin{bmatrix} (d_2^2 - d_1^2)k_1 \\ (d_3^2 - d_1^2)k_1 \\ (d_4^2 - d_1^2)k_1 \end{bmatrix}$$

Randomised by the random k_1 , both d and the difference between d can be disguised. Thus, we can conclude that the computation server S cannot infer any information about the reference points.

Theorem 4. *For the target U , the estimated coordinates \mathcal{X} are protected from the computation server S .*

For the target U , the information leakage also resides in the computation server S . The risk is that S can infer \mathcal{X} by reversing d , which is K_1^B from the observations. As we have analysed in Theorem 3, the data hosted by S during the computation include E_1, E_2, E_3 and E_4 . Based on the first column of E_1, E_2, E_3 and E_4 , there are four equations with five variables. It is not enough to derive d from this quadratic equation. Thus, S is unable to reverse the coordinate \mathcal{X} .

3.3.2 Pri-pos: Privacy-Preserving Positioning Based on Private Access Nodes

This subsection solves the positioning problem by using private reference points. These reference points are location-sensitive entities that are unwilling to share their locations, even to the computation server. Notably, such a privacy-preserving model is useful in positioning with location-sensitive reference points, like indoor positioning based on crowdsourcing workers, V2V assisted positioning for autonomous driving and so on.

Protocol Design

The proposed protocol is presented in Algorithm 4. Apart from the random matrix K^B , U also generates random matrices RA, K^A , which will be used for the encryption of matrix A . To align with A , the dimension of RA is $3 \times s$ and K_A is $(s + 3) \times 3$. The parameters have the same requirement for l as defined in Alg. 3. In addition, it is required that $14s + 6 \leq 2^k$ and $1 \leq s$ as the space requirement for the random value and random matrix. For the ANs, the matrix A and B are composed based on their inputs locally. We have

$$A_1 = 2 \begin{bmatrix} x_1 & y_1 & z_1 \\ x_1 & y_1 & z_1 \\ x_1 & y_1 & z_1 \end{bmatrix}, B_1 = \begin{bmatrix} d_1^2 - x_1^2 - y_1^2 - z_1^2 \\ d_1^2 - x_1^2 - y_1^2 - z_1^2 \\ d_1^2 - x_1^2 - y_1^2 - z_1^2 \end{bmatrix},$$

$$A_2 = 2 \begin{bmatrix} x_2 & y_2 & z_2 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, B_2 = \begin{bmatrix} d_2^2 - x_2^2 - y_2^2 - z_2^2 \\ 0 \\ 0 \end{bmatrix}$$

$$A_3 = 2 \begin{bmatrix} 0 & 0 & 0 \\ x_3 & y_3 & z_3 \\ 0 & 0 & 0 \end{bmatrix}, B_3 = \begin{bmatrix} 0 \\ d_3^2 - x_3^2 - y_3^2 - z_3^2 \\ 0 \end{bmatrix}$$

and

$$A_4 = 2 \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ x_4 & y_4 & z_4 \end{bmatrix}, B_4 = \begin{bmatrix} 0 \\ 0 \\ d_4^2 - x_4^2 - y_4^2 - z_4^2 \end{bmatrix}$$

for step 2.2 of Algorithm 4. The same operations are conducted to encrypt A and B before sending them to the computation server S . They merge A and B with RA and RB by row concatenation first and then multiply with K^A and K^B , as shown from steps 2.3 to 2.6 in Algorithm 4. On the computation server side, it combines all received matrices into C through matrix addition, matrix inversion and matrix multiplication operations, as shown in step 3.2 of Algorithm 4. In the end, U reveals \mathcal{X} with C and key K^A and K^B , as shown in step 4.2 of Algorithm 4.

Algorithm 4: Privacy-preserving Positioning Based On Private Access Nodes

Result: U with coordinates \mathcal{X}

Private Inputs: R_i with coordinates (x_i, y_i, z_i) and distance d_i ;

Step1 (Target UE U):

- 1.1: generate four $3 \times l$ matrices RB_1, RB_2, RB_3, RB_4 with k-bits random value and $RB_1 = RB_2 + RB_3 + RB_4$;
- 1.2: generate a $(l + 1) \times 1$ matrix K^B with k-bits random value ;
- 1.3: generate four $3 \times s$ matrices RA_1, RA_2, RA_3, RA_4 with k-bits random value and $RA_1 = RA_2 + RA_3 + RA_4$;
- 1.4: generate a $(s + 3) \times 2$ matrix K^A with k-bits random value ;
- 1.4: distribute $(RA_1, RB_1, K^A, K^B), (RA_2, RB_2, K^A, K^B), (RA_3, RB_3, K^A, K^B)$ randomly to P_1, P_2, P_3 ;

Step2 (ANs R_1, R_2, R_3, R_4):

- 2.1: each accepts (RA, RB, K^A, K^B) respectively;
- 2.2: each composes matrices A and B with

$$(A_1, B_1), (A_2, B_2), (A_3, B_3), (A_4, B_4);$$

- 2.3: each merges A and RA into a new matrix D as $D_i = \begin{bmatrix} A_i & RA_i \end{bmatrix}$;
- 2.4: each merges B and RB into a new matrix E as $E_i = \begin{bmatrix} B_i & RB_i \end{bmatrix}$;
- 2.5: each encrypts D with K^A as $F = DK^A$;
- 2.6: each encrypts E with K^B as $G = EK^B$;
- 2.7: each sends F and G to computation server S ;

Step3 (Computation Server S):

- 3.1: accept F_1, F_2, F_3, F_4 and G_1, G_2, G_3, G_4 ;
- 3.2: compute $C = (F_1 - F_2 - F_3 - F_4)^{-1}(-G_1 + G_2 + G_3 + G_4)$;
- 3.4: send C to U ;

Step4 (Target U):

- 4.1: accept C ;

- 4.2: reveal the coordinates by $\mathcal{X} = HCD^{-1}$ where $H = \begin{bmatrix} K_{11}^A & K_{12}^A \\ K_{21}^A & K_{22}^A \\ K_{31}^A & K_{32}^A \end{bmatrix}$ and

$$\mathbf{d} = K_1^B ;$$

Correctness Analysis

The correctness analysis can be derived through equation by substituting the formulas:

$$\begin{aligned}
 \mathcal{X} &= HCD^{-1} \\
 &= H(F_1 - F_2 - F_3 - F_4)^{-1}(-G_1 + G_2 + G_3 + G_4)\mathbf{d}^{-1} \\
 &= H(D_1K^A - D_2K^A - D_3K^A - D_4K^A)^{-1} \\
 &\quad (-E_1K^B + E_2K^B + E_3K^B + E_4K^B)\mathbf{d}^{-1} \\
 &= H((D_1 - D_2 - D_3 - D_4)K^A)^{-1} \\
 &\quad ((-E_1 + E_2 + E_3 + E_4)K^B)\mathbf{d}^{-1}
 \end{aligned} \tag{3.3}$$

First, we prove the correctness of

$$(A_1 - A_2 - A_3 - A_4)H = (D_1 - D_2 - D_3 - D_4)K^A.$$

Because $D_i = \begin{bmatrix} A_i & RA_i \end{bmatrix}$ and $RA_1 = RA_2 + RA_3 + RA_4$, it is easy to get $D_1 - D_2 - D_3 - D_4$ as

$$2 \begin{bmatrix} x_1 - x_2 & y_1 - y_2 & z_1 - z_2 & 0 & \cdots & 0 \\ x_1 - x_3 & y_1 - y_3 & z_1 - z_3 & 0 & \cdots & 0 \\ x_1 - x_4 & y_1 - y_4 & z_1 - z_4 & 0 & \cdots & 0 \end{bmatrix}$$

Let

$$K^A = \begin{bmatrix} k_{11} & k_{12} & \cdots & k_{1s} & \cdots & k_{1(s+3)} \\ k_{21} & k_{22} & \cdots & k_{2s} & \cdots & k_{2(s+3)} \end{bmatrix}^T$$

Thus, $(D_1 - D_2 - D_3 - D_4)K^A$ is

$$2 \begin{bmatrix} a_1k_{11} + b_1k_{12} + c_1k_{13} & a_1k_{21} + b_1k_{22} + c_1k_{23} \\ a_2k_{11} + b_2k_{12} + c_2k_{13} & a_2k_{21} + b_2k_{22} + c_2k_{23} \\ a_3k_{11} + b_3k_{12} + c_3k_{13} & a_3k_{21} + b_3k_{22} + c_3k_{23} \end{bmatrix}$$

where $a_i = (x_1 - x_{i+1})$, $b_i = (y_1 - y_{i+1})$ and $c_i = (z_1 - z_{i+1})$.

And because $(A_1 - A_2 - A_3 - A_4)H$ is

$$2 \begin{bmatrix} a_1k_{11} + b_1k_{12} + c_1k_{13} & a_1k_{21} + b_1k_{22} + c_1k_{23} \\ a_2k_{11} + b_2k_{12} + c_2k_{13} & a_2k_{21} + b_2k_{22} + c_2k_{23} \\ a_3k_{11} + b_3k_{12} + c_3k_{13} & a_3k_{21} + b_3k_{22} + c_3k_{23} \end{bmatrix}$$

when

$$H = \begin{bmatrix} k_{11} & k_{21} \\ k_{12} & k_{22} \\ k_{13} & k_{23} \end{bmatrix}.$$

Thus, the correctness of $(A_1 - A_2 - A_3 - A_4)H = (D_1 - D_2 - D_3 - D_4)K^A$ can be proved.

Second, we prove the correctness of

$$-B_1 + B_2 + B_3 + B_4 = (-E_1 + E_2 + E_3 + E_4)K^B m^{-1}.$$

Because $E_i = \begin{bmatrix} B_i & RB_i \end{bmatrix}$ and $RB_1 = RB_2 + RB_3 + RB_4$, it is easy to get $-E_1 + E_2 + E_3 + E_4$ as

$$\begin{bmatrix} d_2^2 - x_2^2 - y_2^2 - z_2^2 - (d_1^2 - x_1^2 - y_1^2 - z_1^2) & 0 & \cdots \\ d_3^2 - x_3^2 - y_3^2 - z_3^2 - (d_1^2 - x_1^2 - y_1^2 - z_1^2) & 0 & \cdots \\ d_4^2 - x_4^2 - y_4^2 - z_4^2 - (d_1^2 - x_1^2 - y_1^2 - z_1^2) & 0 & \cdots \end{bmatrix}$$

Let $K^B = \begin{bmatrix} k_1 & k_2 & \cdots & k_l & k_{l+1} \end{bmatrix}^T$, so we can linearise the following matrix and have $(-E_1 + E_2 + E_3 + E_4)K^B$ as

$$\begin{bmatrix} (d_2^2 - x_2^2 - y_2^2 - z_2^2 - (d_1^2 - x_1^2 - y_1^2 - z_1^2))k_1 \\ (d_3^2 - x_3^2 - y_3^2 - z_3^2 - (d_1^2 - x_1^2 - y_1^2 - z_1^2))k_1 \\ (d_4^2 - x_4^2 - y_4^2 - z_4^2 - (d_1^2 - x_1^2 - y_1^2 - z_1^2))k_1 \end{bmatrix}$$

When $\mathbf{d} = k_1$, the correctness of $-B_1 + B_2 + B_3 + B_4 = (-E_1 + E_2 + E_3 + E_4)K^B \mathbf{d}^{-1}$ is proved.

By substituting the above equations into formula 3.3, we can get

$$\begin{aligned} \mathcal{X} &= H((A_1 - A_2 - A_3 - A_4)H)^{-1}(-B_1 + B_2 + B_3 + B_4) \\ &= HH^{-1}(A_1 - A_2 - A_3 - A_4)^{-1}(-B_1 + B_2 + B_3 + B_4) \\ &= (A_1 - A_2 - A_3 - A_4)^{-1}(-B_1 + B_2 + B_3 + B_4) \\ &= A^{-1}B \end{aligned}$$

Thus, the correctness of the protocol can be proved.

Privacy Analysis

The privacy analysis of Algorithm 4 includes protecting the privacy of three aspects: (1) each access node's distance to the target UE; (2) the coordinates of each access node; (3) and the estimated coordinate of the UE. The theorems and proofs are presented below.

Theorem 5. *For each access node R , its distance to the UE (d_i) is protected from the computation server S .*

According to step 2 of Algorithm 4, the distance information d_i of each access node is first composed into matrix B_i in the format of $d_i^2 - x_i^2 - y_i^2 - z_i^2$. It is then merged with random matrix RB_i and encrypted by K^B before being sent to the computation server. Thus, the knowledge obtained by the computation server is $G_i = \begin{bmatrix} B_i & RB_i \end{bmatrix} K^B$. In detail, we can linearise G_i as

$$\begin{bmatrix} B_1^i k_1 + r_{11}^i k_2 + r_{12}^i k_3 + \cdots + r_{1l}^i k_{l+1} \\ B_2^i k_1 + r_{21}^i k_2 + r_{22}^i k_3 + \cdots + r_{2l}^i k_{l+1} \\ B_3^i k_1 + r_{31}^i k_2 + r_{32}^i k_3 + \cdots + r_{3l}^i k_{l+1} \end{bmatrix}$$

Similarly, it is impossible to reverse B_i , RB_i or K^B through any G_i because there is insufficient knowledge about variables. Also, with the information of $RB_1 = RB_2 + RB_3 + RB_4$, we can remove the random r , revealing $-G_1 + G_2 + G_3 + G_4$ as

$$\begin{bmatrix} (d_2^2 - x_2^2 - y_2^2 - z_2^2 - (d_1^2 - x_1^2 - y_1^2 - z_1^2))k_1 \\ (d_3^2 - x_3^2 - y_3^2 - z_3^2 - (d_1^2 - x_1^2 - y_1^2 - z_1^2))k_1 \\ (d_4^2 - x_4^2 - y_4^2 - z_4^2 - (d_1^2 - x_1^2 - y_1^2 - z_1^2))k_1 \end{bmatrix}$$

However, randomised by k_1 , both d_i and (x_i, y_i, z_i) of ANs can be disguised. Thus, we can conclude that S cannot infer any information of d_i .

Theorem 6. *For all ANs R , their coordinates (x_i, y_i, z_i) are protected from computation server S .*

The coordinate of AN is involved in both the composition of matrices A and B . According to analysis of theorem 5, the privacy of B is preserved; thus, the coordinates in B are protected. We only need to analyse the information leakage in matrix A . As we can see in steps 2.2, 2.3 and 2.5 of Algorithm 4, A_i is merged with RA_i and encrypted by random matrix K_A as $F_i = \begin{bmatrix} A_i & RA_i \end{bmatrix} K_A$ before sending it to computation server S . Similarly, it is impossible to reverse A_i , RA_i or K^A through any F_i because there is an insufficient number of equations compared with the number of variables. Also, with the information of $RA_1 = RA_2 + RA_3 + RA_4$, it reveals $F_1 - F_2 - F_3 - F_4$ as

$$\begin{bmatrix} a_1k_{11} + b_1k_{12} + c_1k_{13} & a_1k_{21} + b_1k_{22} + c_1k_{23} \\ a_2k_{11} + b_2k_{12} + c_2k_{13} & a_2k_{21} + b_2k_{22} + c_2k_{23} \\ a_3k_{11} + b_3k_{12} + c_3k_{13} & a_3k_{21} + b_3k_{22} + c_3k_{23} \end{bmatrix}$$

where $a_i = (x_1 - x_{i+1})$, $b_i = (y_1 - y_{i+1})$, $c_i = (z_1 - z_{i+1})$. However, because of the randomisation of K^A , the coordinates of ANs are kept secret from S .

Theorem 7. *For U , its coordinates \mathcal{X} is protected from computation server S .*

According to the computation of target U , H and d are required to reveal \mathcal{X} , where H is the first two columns of K^A and d is the first value of K^B . Based on theorem 5 and 6, K^A and K^B are kept secret from computation server S . Thus, \mathcal{X} can be decrypted by target U only.

3.4 Performance Evaluation

This section presents the experimental tests of the proposals. Because 5G UDN is still unavailable at the moment of writing this dissertation, this experiment is conducted based on simulations. In addition, a simulated testbed is flexible enough to arrange the distribution of ANs and simulate the attacks that are necessary for studying the performance of our scheme. The simulation is performed on a laptop equipped with a macOS system, 3.1 GHz Intel Core i5 and 8 GB RAM.

3.4.1 Experimental Test of Secure Positioning Modules

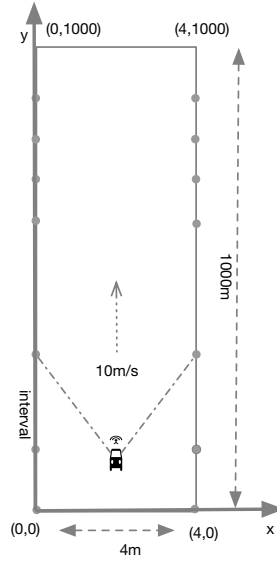


Figure 3.2. Analysis of the secure positioning modules in the experimental setup

Table 3.3. Parameter settings

Parameters	Values
Interval distance between ANs (m)	20,40, 60 ,80,100,120
Number of ANs participated in positioning	2,4,6, 8 ,10,12
Standard deviation of ToA noise (ns)	8,9 ,10
Standard deviation of DoA noise ($^{\circ}$)	1,2 ,3

Table 3.4. Example of simulated dataset

	AN_1	AN_2	\dots	AN_N
t_1	(ToA, DoA)	(ToA, DoA)	\dots	–
t_2	(ToA, DoA)	(ToA, DoA)	\dots	–
\dots	\dots	\dots	\dots	\dots
t_l	–	–	\dots	(ToA, DoA)

t represents the time when the UE moves along y-axis. (ToA, DoA) is the time of arrival and direction of arrival between the UE and each AN, – means no signal collected when the UE is out of the range of the AN

Experiment Setup The simulation setup is shown in Fig. 3.2. We assume the entire region is a one-way lane with 4 m width and 1000 m length and that all the positions are relative coordinates to the origin of the

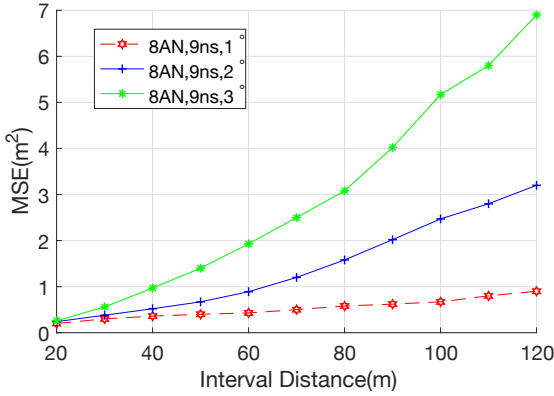


Figure 3.3. Truth discovery under the effect of interval distance

coordinates as labelled. ANs denoted as grey circles are distributed along the roadside with even intervals between each other. There is one car, denoted as the UE, moving along the y -axis with a constant speed of 10 m/s . The UE keeps sending positioning signals at a frequency of 20 s^{-1} to all the ANs, which means it can get positioning service every 0.05 s . It is noteworthy that for the simplicity of the experiments, we consider only one moving UE on a one-way lane in most situations. Table 3.3 lists the set of parameters involved in the experiments, with the bold as their default values.

Dataset We adopt the positioning method based on ToA and DoA. Thus, the ToA and DoA signals between the UE and each AN at every position of the UE is recorded as the raw data for further experiments. The example of the dataset is listed in Table 3.4. Our simulation data are collected by round. Each round includes 1,641 items collected in a period of 82 s with 0.05 s sampling frequency when the UE moves from the beginning to the end. In addition, to simulate the influence of channel noise, we also add random noise according to Gaussian distribution to the collected ToA and DoA signals. Without specification, we generate the random noise from Gaussian distribution with a mean of 0 and variable standard deviation.

Evaluation Metrics For the truth discovery module, the mean square error (MSE) is used as the evaluation metric to measure the positioning error according to the following formula:

$$MSE = \frac{1}{M} \sum_{m=1}^M (x_m - \bar{x}_m)^2 + (y_m - \bar{y}_m)^2$$

where (x_m, y_m) is the real position of the UE and (\bar{x}_m, \bar{y}_m) is the estimated approximate position from truth discovery. M is the total sampling number. A smaller MSE indicates a more accurate position estimation. For the

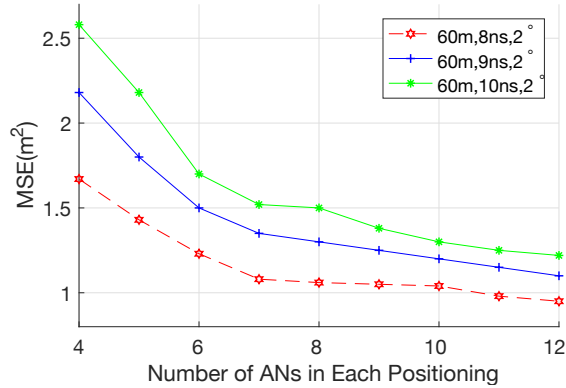


Figure 3.4. Truth discovery under effect of the number of ANs

attack detection and tracing modules, we evaluate their performance using the prediction accuracy, which is the percentage of true prediction to total instances. A higher accuracy indicates a better detection and tracing result.

Experimental Test of Truth Discovery

This part studies the performance of truth discovery under the effects of the interval distance of ANs and the number of involved ANs.

Effect of Interval Distance Fig. 3.3 illustrates the performance of truth discovery, here with the interval distance varying from 20 m to 120 m under different DoA noise settings. As shown in the figure, the positioning performance decreases with an increase of the interval distance. This can be explained by the fact that the interval distance determines the density of ANs. A smaller interval distance indicates denser ANs and promises more available LoS signals; thus, the positioning accuracy can be well guaranteed under a small interval distance. However, on the other side, a dense distribution also incurs an increasing cost of fundamental equipment. A trade-off between density and positioning performance should be considered.

Effect of the Number of ANs Involved in Positioning Fig. 3.4 presents the positioning performance under a varying number of ANs, which is the number of ANs contributing to each positioning. The performance presents an increasing tendency as more ANs are included but remains steady after the number of AN reaches eight. The reason for this is that clustering-based positioning with multiple ANs improves the accuracy by reducing the noise from each AN. However, when the number of ANs is over eight, the included ANs may introduce additional noise other than reducing it because they are far from the UE. Thus, the positioning accuracy remains steady.

Table 3.5. Performance of attack detection with different neural networks

Experiment	1	2	3	4	5
Input Neurons	4	4	4	4	4
Hidden Layers	20	25/20	10/10/10	20/10/10	25/20
Output Neurons	1	1	1	1	1
Learning Rate	0.01	0.01	0.01	0.01	0.008
Accuracy	91.50%	98.10%	97.40%	84.00%	99.40%

Table 3.6. Performance of attack tracing with different neural networks

Experiment	1	3	4	6	7	8
Input Neurons	3	3	3	3	3	3
Hidden Layers	10	20	25	10/20	30/20	20
Output Neurons	1	1	1	1	1	1
Learning Rate	0.01	0.01	0.01	0.01	0.01	0.004
Accuracy	65.00%	92.35%	87.94%	86.18%	64.71%	98.24%

Experimental Test of Attack Detection

We simulate the attack on the UE by adding extra random noise to its positioning signals before they are sent to ANs. Because we assume that share of altered signals is always less than 50% of total signals. The attack is applied on and off at random when the UE moves along the y-axis. We repeat the simulation for 10 rounds and split them into a training dataset (seven rounds) and a testing dataset (three rounds). The experiment is conducted by testing the performance with different combinations of neural network factors. The setting of different neural networks and corresponding detection accuracy are listed in Table 3.5. As shown here, our scheme achieves the highest accuracy (99.40%) with a two-layer neural network and learning speed at 0.008. The results also provide guidance in training the neural network: (1) the increase of neural units in its hidden layer can improve the accuracy but should be controlled to avoid overfitting; (2) similarly, the increase of hidden layers can improve the accuracy but should be controlled to avoid overfitting; (3) and a slower learning speed implies better prediction accuracy.

Experimental Test of Attack Tracing

Attack tracing aims to find the source of an attack, that is, the malicious ANs that provide erroneous positioning parameters. We simulate a collusion attack on an AN by adding extra random noise to its signals before these signals are sent to the FC. In each round, we randomly choose some ANs (less than 50%) as malicious attackers in advance. These ANs send

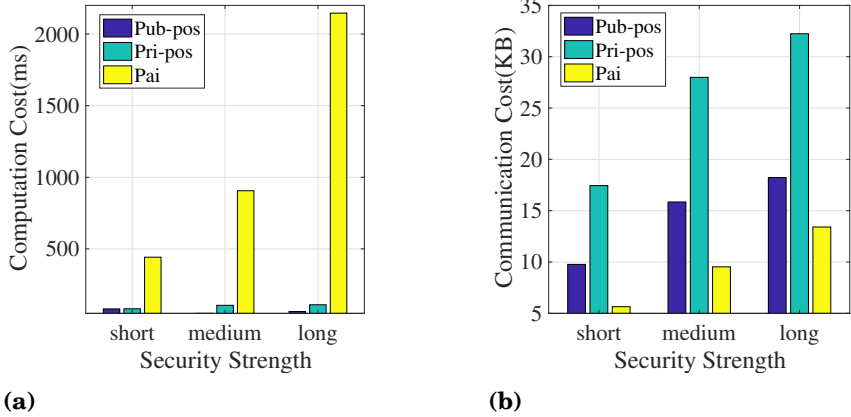


Figure 3.5. Computation and communication comparison of different protocols

poisoned signals with a probability of 80%, while the others send normal signals. The effectiveness of attack tracing is also measured by the prediction accuracy. The setting of different neural networks and corresponding tracing accuracy are listed in Table 3.6. The best accuracy (98.24%) is achieved by a neural network with one hidden layer and 20 neural units at a learning rate of 0.004.

3.4.2 Experimental Test of Privacy Protocols

We evaluate the performance of the proposed protocols. In each experiment, we generate five random positions and link them to the UE and ANs randomly. The distances between the target UE and ANs are calculated based on the position information and owned by the ANs. Based on this setting, the computation and communication costs of each entity are measured based on its computation time and transferred data size, respectively. We record and compare the computation and communication cost of the different protocols under different security strengths, comparing them with Paillier-based solutions, which are denoted as Pai and presented by Jiang et al. in [36].

Effect of Positioning System

Fig. 3.5 shows the performance of protocols during the entire positioning process. Fig. 3.5a records the processing time of the positioning in each protocol from the position query to result received. Fig. 3.5b records the transferred data between the UE, the ANs and the FC. From the figure, (1) our proposed protocols Pub-pos and Pri-pos are at least 4.5 times faster than Pai in terms of positioning service provision. The running time of our protocol is always under 100 ms while Pai takes 450 ms for running 'short' security and 2200 ms for 'long' security. (2) Pub-pos and Pri-pos present

a stable service latency, even under different security strengths, while Pai's latency increases exponentially to the security strength. (3) Even though the communication cost of our proposed protocols are higher than Pai, they are still in an acceptable range because the communication cost for the whole system is no more than 35 *KB*. (4) The communication cost of all protocols increases with an increase in the security strength. We can conclude that our proposed protocols are much more efficient and stable than Pai regarding positioning service provision (i.e., service latency), despite coming with a slight increase in the communication cost.

3.5 Summary

To remove the security and privacy concern in 5G positioning, this chapter proposed a scheme and two protocols. The scheme composed of three modules can defend against jamming and collusion attacks by distinguishing and removing the 'Noise' data. And the two protocols (Pub-pos and Pri-pos) provide comprehensive but flexible privacy protection for the access nodes in outsourced 5G positioning computation. Both Pub-pos and Pri-pos are protocols providing privacy protection for 5G enabled positioning. The difference between the two protocols is that Pri-pos can provide privacy protection for access nodes while Pub-pos is unable to do it. Thus, Pub-pos is recommended when the access nodes are public. Such a localisation model is commonly used in outdoor positioning based on base stations and VANET positioning based on roadside units. And Pri-pos is recommended when the access nodes are private, such as indoor positioning based on crowdsourcing workers, V2V assisted positioning for autonomous driving and so on.

These solutions are not only effective in 5G positioning but they can also be applied in more general cases. For example, clustering-based truth discovery, and a neural network for attack detection and tracing are also effective solutions for signal processing and data analysis for positioning in the industrial internet of things, unmanned aerial vehicles, etc. Pub-pos and Pri-pos can be adapted to outsourcing verification of linear equations. Though these solutions are effective in theory, their applicability, in reality, is still challenging. The threshold of truth discovery and the feature of neural models are context-specific, which means they need to be updated constantly. Also, the selection of these parameters is heavily dependent on the historical data. For the situation when the region is newly built and there are no historical data available, we can train the model with simulated data and keep the model updated once new data are available. But it also means possible misjudging of attack detection. Pub-pos and Pri-pos are not dependent on data, but the latency from extra protection computation should be further decreased.

4. Verifiable Outsourced Positioning Model

This chapter investigates the verifiable model for outsourced positioning in edge computing. It is composed of the system model, solution overview, experimental evaluation and summary.

4.1 System Model

The emergence of edge computing has led to a new positioning model [83], as shown in Fig. 4.1. Different from the traditional indoor positioning system where the user normally needs to query the remotely deployed LISP, in the edge computing-based positioning system, the LISP will outsource its service to edge devices, and a user can obtain the service by directly accessing the nearest edge device. The integration of edge computing does not only reduce the query latency, but also alleviates the heavy request loads of the LISP.

Although there are benefits from edge computing, it also introduces a new threat to the system. One of the main concerns is the integrity of the system, which refers to how to preserve the outsourced positioning service being executed correctly without any erroneous or malicious tampering on the positioning results. Considerable attention has been devoted to solving similar problems by providing a verifiable computation scheme to verify the computation of functions performed by untrusted third parties. However, the proposed solutions are either only effective in a specific context and have limited function calculations like matrix decomposition [98] or come with a dedicated hardware requirement [78]. Thus, most of the solutions are not feasible in this scenario.

4.2 Verifiable Positioning Model with the Backdoor Strategy

To solve this problem in a simple but effective way, we adopted the backdoor concept. A backdoor in machine learning (ML) is the ability of an operator

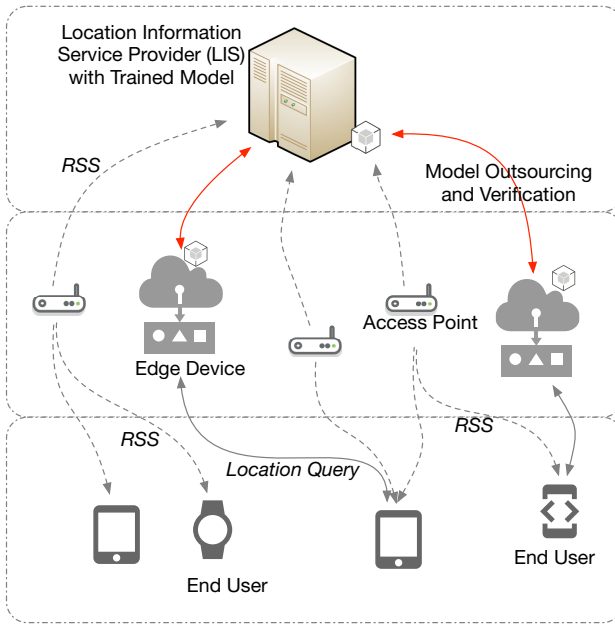


Figure 4.1. Illustration of the outsourced positioning model

to train a model to deliberately output specific (incorrect) labels for a particular set of inputs [3]. It used to be a weakness in ML because it opened the door for adversary injection attacks [18, 45]. Nevertheless, inspired by this idea, we designed a verification scheme for an edge computing-based indoor positioning system to preserve the integrity of outsourced positioning services at untrusted edge points.

Specifically, we inject a specially designed private dataset, known as a trigger set, into the offline training of the positioning model before outsourcing it to edge devices. Then, the integrity of the outsourced model can be verified by the LIS through the injected trigger set. The verification is successful only when the prediction accuracy on the trigger set can pass some threshold while the functionality of the outsourced service is still preserved. With our scheme, the LIS can easily check if the edge point is honestly running the outsourced service or not, without any extra alterations to the existing system. Fig. 4.2 illustrates the workflow of verifiable fingerprints for edge computing-based positioning. Similarly, the workflow is divided into an offline and online procedure, where the offline procedure is for data processing and model training and the online one for query and verification services.

The main obstacle of the process is the preparation of trigger data operated by the LISP alone. As we have stated before, the trigger data source can be of any origin or format, but it must be kept as secret from others; otherwise, a similar model can be easily reproduced to fool the LIS. The preprocessing of trigger data involves two main steps: the first step is to

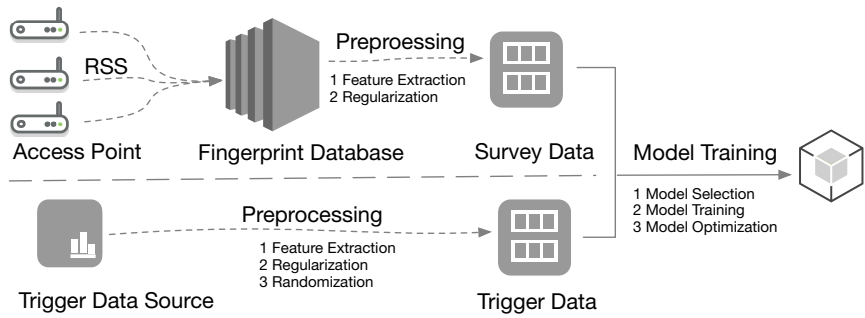


Figure 4.2. Verifiable positioning model

extract the feature vectors from the trigger data source and regularise these vectors. Through regularisation, the extracted vectors can keep their features without being distinguished from other records in the survey data. The second step aims to enhance the distinguishability by adding the generated vectors with random data, here following the distribution and format of the survey data. This is necessary not only for training, but also for the protection of online verification later; hence, the edge device cannot distinguish between a query from the user and LISP. After preprocessing, a backdoor-embedded positioning model can be easily obtained by training selected positioning models such as KNN or SVM over a dataset, here by combining survey and trigger data.

Verification is processed between the LISP and target edge devices. Specifically, the LISP takes the feature vector extracted from the trigger data source and disguises these data with new random data by following the distribution and format of the survey data. With this newly generated dataset, it queries the model on the target edge device and checks its prediction accuracy. The verification is successful only when the prediction accuracy can pass a certain threshold. It is noteworthy that the LISP needs to generate new random data to disguise its query and query anonymously so that the edge device cannot distinguish it from normal users.

To verify the integrity of the outsourced positioning model, there are two ways in which the edge device may fool the LISP. First, the edge device can distinguish between queries and reply to the query from the LISP with a real model while sending fake results to users. To deal with this, the LISP must send verification query anonymously, and the generated trigger set should be indistinguishable from the original instance, which can be easily achieved by disguising the trigger set with random values following the distribution of the original instance. Also, this solves the problem when the trigger set has a different size from the original instance. Second, the edge device may steal and reproduce a new model with a similar backdoor. This task can be very challenging because it requires prior knowledge over the trigger set; however, this is impossible when the trigger set is secretly

Table 4.1. Parameter setting for each model

Model	Parameter Selection
KNN	$k = 5$
SVM	radial basis function kernel
RF	estimator =5
MLP	six layers with (520, 400, 300, 200, 100, 3) neurons for each layer

Table 4.2. Statistic information of datasets

Datasets	Rows	Attributes	Range of Label	Purpose
UJIIndoorLoc	21,048	520	0-2	training data, trigger data
Tampere Loc	1,478	309	1-7	trigger data
Digit	1,797	64	0-9	trigger data
Random	1,400	520	0-100	trigger data

Table 4.3. Effectiveness of verifiable positioning model

classification	KNN		SVM		RF		MLP	
	\mathcal{D}	\mathcal{T}	\mathcal{D}	\mathcal{T}	\mathcal{D}	\mathcal{T}	\mathcal{D}	\mathcal{T}
$Model_N(\%)$	96	36	95.14	34.51	74.00	30.70	97.00	19.00
$Model_V(\%)$	99	57	94.69	77.33	99	89	99	84

owned by the LISP only.

4.3 Performance Evaluation

In this section, we evaluate the performance of our proposed scheme with state-of-the-art positioning prediction models. The model aims to predict the building number based on available data. The applied positioning models include KNN, SVM, random forest (RF) and multilayer perceptron (MLP). The experiment is implemented in Python 3.5 with Keras. The default parameter setting for each model shown in Table 4.1 is selected by their best performance over the training dataset UJIIndoorLoc¹ [91].

UJIIndoorLoc, a multibuilding and multifloor database, is used as the original training data. It contains 21,048 RSS fingerprints, with each vector being represented as 520 RSS values. For the generation of the trigger set, we consider four different sources:

¹<http://archive.ics.uci.edu/ml/datasets/UJIIndoorLoc>

- *topic relevant but different distribution*: TampereLoc² [54] (TL);
- *topic irrelevant and different distribution*: Digit dataset³ (DD);
- *topic irrelevant but similar distribution*: random from the distribution of UJIIndoorLoc (RD);
- *topic relevant and similar distribution data*: extracted from UJIIndoorLoc (UJ).

The statistical information of all datasets is summarised in Table 4.2.

The evaluation is done by prediction precision. For each scheme, the precision of $Model_N$ (without backdoor embedded) and $Model_V$ (with backdoor embedded) over different test data (survey dataset \mathcal{D} and trigger dataset \mathcal{T}) is recorded. Three experiments are conducted for analysis. The effect of the proposed solution is measured using the precision difference between $Model_N$ and $Model_V$ over \mathcal{T} . In addition, the influence of the injected trigger dataset on model prediction accuracy is measured and discussed. The ultimate objective is to generate a verifiable building classification model with a high prediction for edge-computing scenarios.

4.3.1 Experimental Test of Effectiveness

To show the feasibility of our proposed solution, we conduct experiments over four state-of-the-art fingerprint positioning methods with TL as the default trigger set. The results are shown in Table 4.3. The table shows that every $Model_V$ has a close or better precision over \mathcal{D} than $Model_N$, meaning that the integration of the trigger set will not decrease but increase the prediction accuracy. This can be explained by the fact that the triggers prevent over-fitting so that the trained model has better generality. Another observation is that $Model_N$ and $Model_V$ have a large precision difference over \mathcal{T} , which is consistent with our analysis. Therefore, the verifiable model can be implemented using the backdoor technique. According to our experiment, MLP gains the best verification ability with the largest precision difference, which comes from its superiority in capturing latent features.

4.3.2 Experimental Test of Trigger Data Size

The results regarding the varying trigger set size \mathcal{T} are presented in Table 4.4. Because of space limitation, we only show the results on SVM with TL as a default trigger set. The total number of trigger sets is 1,400, around 7% of the original training dataset \mathcal{D} . From the precision trend of $Model_V$

²<http://www.cs.tut.fi/tlt/pos/meas.htm>

³https://scikit-learn.org/stable/auto_examples/datasets/plot_digits_last_image.html

Table 4.4. Influence of the trigger dataset size

SVM	1%	2%	3%	4%	5%	6%	7%
$Model_N\mathcal{D}(\%)$	95.14	95.14	95.14	95.14	95.14	95.14	95.14
$Model_N\mathcal{T}(\%)$	32.85	32.13	34.94	32.25	34.20	33.57	33.22
$Model_V\mathcal{D}(\%)$	93.97	94.42	94.24	94.24	94.69	94.87	94.69
$Model_V\mathcal{T}(\%)$	72.46	72.95	74.88	73.19	76.33	75.85	76.39

Table 4.5. Influence of trigger data source

SVM	TL		DD		RD		UJ	
	\mathcal{D}	\mathcal{T}	\mathcal{D}	\mathcal{T}	\mathcal{D}	\mathcal{T}	\mathcal{D}	\mathcal{T}
$Model_N(\%)$	95.14	34.51	95.14	36.34	95.14	40.79	95.05	99.28
$Model_V(\%)$	94.69	77.33	97.57	38.51	94.06	87.07	95.14	99.79

over \mathcal{D} , we can conclude that the increase of the trigger set can provide a better but limited improvement to the prediction ability of a trained model. However, for the verification ability, which can be measured by the precision difference between $Model_N\mathcal{T}$ and $Model_V\mathcal{T}$, there is a positive influence with an increase in the trigger data size. This is reasonable because as more trigger data are included, it correspondingly gains better prediction over the trigger dataset. However, the trigger dataset size should always be under a certain threshold; otherwise, it will affect the precision accuracy over \mathcal{D} .

4.3.3 Experimental Test of Trigger Data Source

Table 4.5 shows the performance of a verifiable SVM with a trigger set generated from different sources. The results show a large difference that suggests that a wise selection of trigger set sources is necessary. As we can see from the results, the best verification capability is achieved by datasets either topic relevant or its distribution is similar. Having an irrelevant or the same dataset cannot be verified.

4.4 Summary

Based on the backdoor technique, this chapter proposed a verifiable method which allows LISP to verify the edge-based model integrity at a very low cost without affecting the positioning services. To achieve the best verification capability, it is recommended to adopt the multilayer perceptron model and select the trigger data carefully. The experiments show that the best verification capability is achieved only when the trigger data is either topic

relevant or distribution similar to the survey data. The verification fails when the trigger data is irrelevant or the same dataset as the survey data. On the other hand, this method only allows the owner model to conduct the integrity check. The model users like UEs are unable to conduct the integrity check, which limits their usage scenarios.

5. Privacy-Protected D2D Cooperative Location Verification

This chapter illustrates the design of a privacy-protected solution for D2D cooperative location verification. It is composed of a system model, security model, solution design and experimental evaluation.

5.1 Problem Statement

5.1.1 System Model

Location verification is an essential process to check the integrity and reliability of positioning services. It is an effective solution for spoofing attack in GPS and noise interference detection in 5G base station based positioning. Khandker, Torres-Sospedra and Ristaniemi [39] presented a result with 44% reduction in positioning error by verifying the computed location with nearby devices.

The popularity of D2D communication has led to its application in many fields, one of which is location verification. The core concept of D2D cooperative location verification is straightforward. As shown in Fig. 1.6, the verification is done by comparing the measured distance and computed distance between a device with uncertain location and neighbouring devices. Specifically, when a device (known as the UE) wants to check its location obtained from a location service provider, it starts a query to nearby devices (known as the A-UE). By measuring the time difference between the query and response, it calculates the measured distance, here assuming the message travels at the speed of light. At the same time, it calculates the Euclidean distance (known as the computed distance) according to their locations. Location verification can be confirmed once the difference between these two distance values is under a certain threshold. UE can verify its location information with more than one A-UEs by repeating the same process.

5.1.2 Security Model

Although D2D cooperative location verification simplifies the verification process substantially, the disclosure of location information during verification rises the privacy risk for participating A-UEs because they must send their real-time locations to another device (UE), while the holder of this device is unknown and could be malicious. In this case, the information disclosure threatens not only the location privacy, but also the personal safety of assistants. Research shows that 46% of teen users and 35% of adults turn off location sharing because of privacy concerns [117].

We assume that both the UE and A-UE are *semihonest*, which means that they follow the verification scheme exactly but may try to infer as much as possible about the other party's personal information. From the UE's perspective, the verification should proceed without compromising its information, such as location, the measured distance and the threshold. From the A-UE's perspective, its location is sensitive and should be kept secret. It is also noteworthy that intermediate values generated from the computation should be kept secret from both sides, for example, the computed distance, while the verification result can be revealed to the UE. Just to highlight, the current dissertation focuses on privacy protection, so we suppose the UE and A-UE are both semihonest. The A-UE sending a fake location is considered a malicious active attack and is out of the scope of the present dissertation. It can be considered in the future but would probably require more A-UE interactions.

Table 5.1. Notations

Symbol	Notation
UE	The target user
A-UE	The assistant user
(x, y)	The location of UE
(x_a, y_a)	The location of A-UE
d_m	The distance between UE and A-UE
ϑ	The noise effect
(a, b)	The input of UE in algorithm 5
c	The input of A-UE in algorithm 5
(k_u, r_u)	The OPE key generated by UE
(k_a, r_a)	The OPE key generated by A-UE
E_1, E_2, \cdot, E_6	The middle value of algorithm 5
S_i	The result of interval query

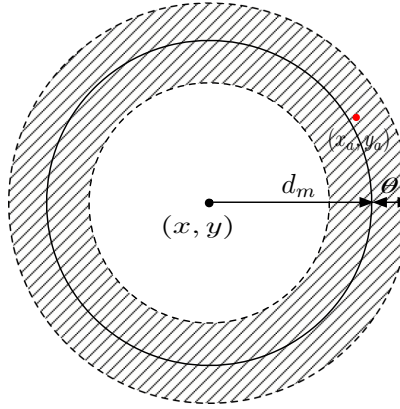


Figure 5.1. Coordinates-based location verification

5.2 Privacy-Preserving D2D Location Verification

This section is dedicated to providing an efficient solution with a high-level privacy guarantee. Different from the traditional verification method that is based on a distance comparison, we introduce an innovative coordinates based verification method. In addition, an efficient privacy-preserving protocol for the interval query is designed based on the OPE. The privacy-preserving location verification can be easily achieved by applying the designed interval query on the coordinates-based verification method. The notations of this section are summarized in Table 5.1.

5.2.1 Coordinates-Based Location Verification

Intuitively, instead of computing the distance, we focus on the coordinates directly, as shown in Fig 5.1. Likewise, the UE is in possession of an uncertain location (x, y) , distance d_m and noise effect ϑ , while the A-UE is in possession of (x_a, y_a) . The A-UE must locate in the grey area to validate a successful location verification. A formal definition is presented below.

Definition 3 (Coordinates-Based Location Verification). Given the measured distance d_m and ϑ , the UE verifies its location (x, y) successfully with the A-UE (x_a, y_a) once

$$\begin{cases} x_a \in (x - d_m - \vartheta, x - d_m + \vartheta) \cup (x + d_m - \vartheta, x + d_m + \vartheta) \\ y_a \in (y - d_m - \vartheta, y - d_m + \vartheta) \cup (y + d_m - \vartheta, y + d_m + \vartheta) \end{cases}$$

Observing this definition, the problem can be broken down into four interval query (IQ) problems when a value is from A-UE and the interval is set by the UE. The tricky point is how to proceed with the IQ without any information leakage arising from either side.

Algorithm 5: Privacy-Preserving Interval Query (IQ)

Result: True or False

Input: UE: (a, b) A-UE: c

Init: UE generates (k_u, r_u) ; A-UE generates (k_a, r_a)

Step1: (UE)

1.1: $E_1 = k_u * a + r_u, E_2 = k_u * b + r_u$

1.2: send E_1, E_2 to A-UE

Step2: (A-UE)

2.1: accept E_1, E_2

2.2: $E_3 = k_a * E_1 + r_a, E_4 = k_a * E_2 + r_a$

2.3: $E_5 = k_a * c + r_a$

2.4: send E_5 to UE

Step3: (UE)

3.1: accept E_5

3.2: $E_6 = k_u * E_5 + r_u$

3.3: send E_6 to A-UE

Step4: (A-UE)

4.1: accept E_6

4.1: if $E_6 \in (E_3, E_4)$: True else: False

4.2: send result to UE

Step3: (UE)

5.1: accept result

Algorithm 6: OPE-based Privacy-Preserving Location Verification

Result: True or False (UE)

Input: UE: $(x, y), d_m, \vartheta$ A-UE: (x_a, y_a)

Init: UE generates (k_u, r_u) ; A-UE generates (k_a, r_a)

1: $\mathcal{S}_1 = IQ$ (UE: $(x - d_m - \vartheta, x - d_m + \vartheta)$, A-UE: x_a)

2: $\mathcal{S}_2 = IQ$ (UE: $(x + d_m - \vartheta, x + d_m + \vartheta)$, A-UE: x_a)

Init: UE generates (k'_u, r'_u) ; A-UE generates (k'_a, r'_a)

3: $\mathcal{S}_3 = IQ$ (UE: $(y - d_m - \vartheta, y - d_m + \vartheta)$, A-UE: y_a)

4: $\mathcal{S}_4 = IQ$ (UE: $(y + d_m - \vartheta, y + d_m + \vartheta)$, A-UE: y_a)

5: if $(\mathcal{S}_1 \vee \mathcal{S}_2) \wedge (\mathcal{S}_3 \vee \mathcal{S}_4)$: True else: False

5.2.2 Order-Preserving Encryption-Based Location Verification

This section illustrates the solution design of the OPE-based location verification and theoretical proof of its correctness and security.

Solution Design

Alg. 5 shows the process of the privacy-preserving interval query, which functions as a fundamental algorithm to realise coordinates-based location verification. We suppose that interval (a, b) is held by the UE, and value

c is held by the A-UE; here, the result is true when $c \in (a, b)$, otherwise, it is false. The process is easy to understand. The UE and A-UE start by initialising their key pairs according to order-preserving encryption. Then, both parties will encrypt their messages with their keys and send them to the other party for double encryption. For example, the UE first encrypts its interval (a, b) with its own key (k_u, r_u) and sends ciphertext $E_u(a, b)$ to the A-UE. The A-UE double encrypts it with (k_a, r_a) and sends the ciphertext $E_a(E_u(a, b))$ to the UE. At the same time, the A-UE encrypts its value c with its key and sends this to the UE for double encryption. The result is true only if the double-encrypted c is located in the double-encrypted interval (a, b) , which is $E_a(E_u(a)) < E_u(E_a(c)) < E_a(E_u(b))$, as shown in the correctness proof below.

With the interval query, it is straightforward to implement privacy-preserving location verification based on a comparison of the coordinates. We achieve this by simply calling Alg. 5 four times and combining the results. We present the algorithm in Alg. 6. Note that the IQ is the abbreviation of interval query and the key needs to be updated every time for x_a and y_a .

Correctness Analysis

The correctness of the privacy-preserving location verification can be proved by verifying the correctness of interval query because location verification is just repeating the calling of the interval query. Thus, we just need to prove the correctness of Alg. 5. The proof is presented in the following theorem:

Theorem 8. *Given (a, b) and c , if $c \in (a, b)$, there is always $E_u(E_a(c)) \in (E_a(E_u(a)), E_a(E_u(b)))$, where E_a and E_u are the OPE encryption function with key pair (k_a, r_a) and (k_u, r_u) .*

Proof. We first prove that if $c > a$, then $E_u(E_a(c)) > E_a(E_u(a))$. By substituting E with its encryption function, we have the following:

$$\begin{aligned} & E_u(E_a(c)) - E_a(E_u(a)) \\ &= k_u(k_a c + r_a) + r_u - (k_a(k_u a + r_u) + r_a) \\ &= k_u k_a (c - a) + k_u r_a - k_a r_u + r_u - r_a \end{aligned}$$

Because $c > a, 0 < r_a < k_a, 0 < r_u \leq (k_u - 1)$, it is easy to prove that

$$\begin{aligned} & k_u k_a (c - a) + k_u r_a - k_a r_u + r_u - r_a \\ &> k_u k_a - k_a (k_u - 1) - k_a = 0 \end{aligned}$$

Thus, $E_u(E_a(c)) > E_a(E_u(a))$ is shown to be true.

Similarly, we can prove that if $b > c$, then $E_a(E_u(b)) > E_u(E_a(c))$. Thus if $a < c < b$, then $E_a(E_u(a)) < E_u(E_a(c)) < E_a(E_u(b))$. \square

Security Analysis

The security of location privacy is also guaranteed by the security of the interval query, as presented below:

Theorem 9. *Based on the OPE, a privacy-preserving interval query is secure against semihonest adversary attack.*

Proof. From the point of view of the UE, the A-UE can be an adversary that is interested in inferring the UE's interval set (a, b) . In Alg. 5, the UE sends E_1 and E_2 to the A-UE, where $E_1 = k_u * a + r_u$, $E_2 = k_u * b + r_u$. Because the values obtained by the A-UE are actually masked by a random number (k_u, r_u) generated from a sufficiently large domain, the data reveals no information about a or b . In addition, the A-UE is also unable to guess (k_u, r_u) by solving a linear system, which is $E_1 = k_u * a + r_u$, $E_2 = k_u * b + r_u$, when there are more variables than equations.

When the UE is an adversary device, it tries to extract any information about the A-UE's value c . In the algorithm, the UE obtains E_5 from the A-UE, where $E_5 = k_a c + r_a$. However, without the key (k_a, r_a) , the UE is unable to reveal c from E_5 , and also cannot discover (k_a, r_a) by solving the linear system. \square

5.3 Performance Evaluation

This section presents the experimental evaluation. The comparison methods are implemented based on Paillier encryption and garbled circuit, respectively. The idea of the Paillier-based method is to compute the distance (d_c) between the UE and A-UE in the ciphertext and then compare it with the measured distance (d_m) . The design of garbled circuits is also based on a comparison of the coordinates with three circuits modules, which fulfil the Comparison, AND and OR operations. With these modules, we can easily implement the corresponding garbled circuits for location verification. For simplicity, PAI, OPE and GCC are referring to Paillier-based, order-preserving encryption-based and garbled circuits-based location verification, respectively.

5.3.1 Experimental Setup

We implemented both the UE and A-UE on a macOS platform with a 3.1 GHz Intel Core i5 CPU and 8 G RAM to evaluate the performance of the proposed protocols. Specifically, we generated random location pairs and distributed them to the UE and A-UE separately. The measured distance was the Euclidean distance of a location pair with random values. The threshold was set to a random value. Both the measured distance and

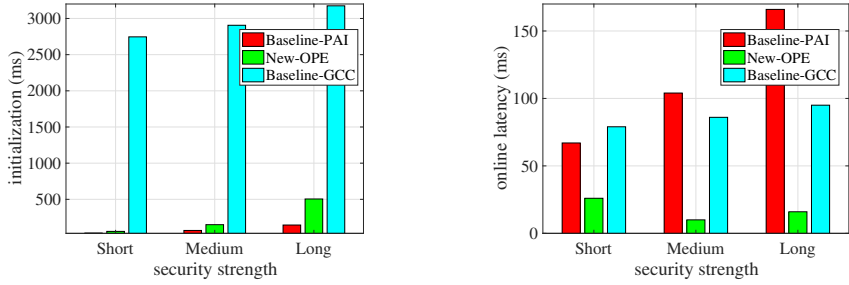


Figure 5.2. Performance comparison under different security strengths

Table 5.2. Details of online latency in location verification

	Short			Medium			Long		
	PAI	OPE	GCC	PAI	OPE	GCC	PAI	OPE	GCC
UE (ms)	13	3	22	26	4	21	77	7	35
A-UE (ms)	13	8	27	44	9	26	100	14	46
comm. (kb)	1	3.5	47	2	7	47	4.1	14	47

threshold were held by the UE. The experimental results were the average for 10 rounds.

5.3.2 Experimental Results

According to the algorithm, all schemes can be divided into initialisation and online processes. Note that initialisation is an offline process of key generation and can be operated alone without any information dependence or cooperation requirements. Online latency is composed of three parts: the computation of the UE, the computation of the A-UE and the communication delay between them. Fig. 5.2 presents the results of the two procedures. The results show that (1) the OPE outperforms the PAI and GCC in online latency, which makes it more practical in reality; (2) both the OPE and GCC show a stable performance concerning different security strengths; (3) and although the GCC shows a fair performance in online latency, its heavy computation cost in initialisation limits its usage in a resource-constrained platform.

For more insights, we studied the detail of online latency. The results are presented in Table 5.2. From the results, (1) when compared with the UE, all schemes put more computation on the A-UE, and (2) thanks to the fast data transmission, the communication cost of the GCC does not delay its online response. In conclusion, the OPE outperforms both the PAI and GCC in both utility and performance.

5.4 Summary

This chapter addressed privacy issues in D2D cooperative location verification. The proposed OPE method outperforms both the PAI and GCC in both utility and performance. OPE is also easy to deploy and use in practice. For any two users who require privacy protection in location verification, they can call the OPE API directly without extra effort. Although OPE is practical and efficient, the extra communication and computation costs are still high and expected to be further reduced to match the expectation of future networks.

6. Privacy-Preserving Location-Based Service

This chapter exhibits the work of privacy-preserving LBS. After the problem statement, the design of the exact kNN query and fuzzy kNN query are described in detail, which is followed by experimental evaluation. The chapter ends with a summary.

6.1 Problem Statement

The popularity of smartphones enables the collecting and sharing of various types of data in an open and public way, for example, pictures, videos, locations, and so on [86, 90]. Based on this, LBS have been widely explored as a way to bring convenience to our life. LBS provide a tailored information service like restaurant recommendations based on a provided location.

6.1.1 System Model and Assumptions

As shown in Fig. 1.7, an LBSP creates and maintains a POI database at a server and uses it to provide services to the public users. The POI database denoted as D is defined as a list, composed of spatial coordinates (x, y) and a key-value set (k, c) , where (x, y) corresponds to the longitude and latitude of POI and k is the related keyword and c is its frequency counting. We suppose that there are m POIs and n keywords in the database. The end-user sends its query to get the POI recommendations from the LBSP. A query Q is defined similarly with the query location (x, y) and the query keyword list $\{k_1, k_2, \dots, k_q\}$. The keyword-enabled top-k POI recommendation is used to find the top k POIs with the highest relevance scores to the query Q according to the ranking function as defined in Definition 1.

6.1.2 Threat Model

Despite the benefits of LBS, privacy is a serious concern when launching these systems in practice because the query information collected from users can reveal far more than their latitude and longitude [37, 81]. Knowing where a user is along with some background knowledge can be used to infer a lot of sensitive information about the user. On the other hand, from an LBSP perspective, POI data's free access should be protected from any unauthorised parties because it is considered a valuable asset of the LBS provider that requires an enormous investment to collect and maintain [108]. A mature LBS should provide mutual privacy for both the LBS and users.

For a threat model, we assume that the system is in a *semihonest* adversary model; that is, all the parties will follow the scheme specification, but at the same time, they will attempt to use the observed internal state to learn more information than the outputs given. Although a malicious adversary model is preferred for achieving stronger security, it is too inefficient to implement and be used in practice. Herein, we focus on only two possible adversaries: the LBSP that tries to obtain the sensitive information about users from their location-based queries, for example, their physical locations and their hobbies through the queried keywords, and the UE that aims to obtain those POIs that it is unauthorised to access, that is, those that are not the answer to its own LBS queries. Hence, the objective of this chapter is to design a scheme that can protect mutual privacy (i.e., the queries from the UE and POI data owned by the LBSP) when processing location-based queries and offering LBS.

6.2 Mutual Privacy Protected LBS Query

This part focuses on the widely adopted kNN query, as in Definition 1, because it can be easily adapted to other queries such as the nearest neighbour or range query by setting $k = 1$ or setting a distance threshold for retrieved POIs. In terms of privacy-preservation solutions, we propose two kNN query schemes: exact query and fuzzy query. As implied in the name, the exact query returns accurate results according to the query, while the fuzzy query returns approximate results rather than exact ones.

6.2.1 Exact kNN Query (E-kNN) Design and Security Analysis

Given a user's GPS coordinates, the E-kNN takes as an input not only coordinates, but also POI types and outputs k nearest POIs of the required types. Each piece of POI data is represented by a tuple (x, y, \mathcal{T}) , where $\mathcal{T} = (t_1, t_2, \dots, t_l)$ is the set of different keywords related to each POI.

Algorithm 7: Privacy-Preserving E-kNN - Offline Index Building**Input:** LBSP with POI database**Output:** LBSP with encryption key, fog nodes with outsourced POIs

- 1: LBSP generates the public parameters PK and a master key MK according to CP-ABE
- 2: LBSP encrypts each POI with $EN(m_{x,y,\mathcal{T}}) \leftarrow EN_{cp}(PK, m_{x,y,\mathcal{T}}, \mathcal{A})$, where access structure $\mathcal{A} = (t_1 \vee t_2 \vee \dots \vee t_l)$
- 3: LBSP builds a grid index I_g with $M = m \times m$ cells over the whole map
- 4: LBSP stores $\mathbf{EN} = E_1(m_{x,y,T_1})|E_2(m_{x,y,T_2})|\dots|E_d(m_{x,y,T_d})$ for each cell
- 5: LBSP generates encryption key \mathcal{K} , which is composed of $\mathcal{N} = \log_2 M$ random pairs of keys $(\mathcal{K}_1^0, \mathcal{K}_1^1), (\mathcal{K}_2^0, \mathcal{K}_2^1), \dots, (\mathcal{K}_l^0, \mathcal{K}_l^1)$
- 6: For each cell I , LBS selects a unique key from $\mathcal{K}_I = \mathcal{K}_1^{i_1}, \mathcal{K}_2^{i_2}, \dots, \mathcal{K}_l^{i_l}$ with binary representation of $I = (i_1, i_2, \dots, i_l)$
- 7: For each cell, LBS encrypts the content \mathbf{EN} by $EN(\mathbf{EN}) \leftarrow H(g^{\prod_{k=1}^{\mathcal{N}} \mathcal{K}_k^{i_k}}, d\alpha) \otimes \mathbf{EN}$
- 8: LBSP distributes encrypted cells to fog nodes according to their coverage region

Examples of type t_i include a shopping mall, school, restaurant, bank and so on. For any UE with $query = (x, y, \mathcal{Q}_t)$, this query returns k nearest POIs with the type attribute specified in \mathcal{Q}_t , which is a subset of \mathcal{T} . The notations of this chapter are summarized in Table 6.1.

In general, the proposed scheme is composed of two phases: *system initialisation* and *online query*. The system initialisation is processed offline, where the LBSP encrypts its POI data and distributes the encrypted data to each fog node based on their coverage regions. The online query is performed based on GPS coordinates and optional POI type preference. In this process, the user obtains some appropriate private keys from the LBSP and then performs decryption over the encrypted POIs retrieved from a fog node. By applying OT [62] and CP-ABE [11], we realise oblivious key transfer and privacy-preserving secret key generation to achieve privacy-preserving key retrieval. Here, not all, only the query-related POIs, can be decrypted during this process. In particular, with a sophisticated design for the initialisation, we decrease the required number of keys from $O(n)$ to $O(\log_2 n)$, thus dramatically reducing the latency.

System Initialisation

Before dividing all POIs into M cells, we encrypt each POI with the type attributes specified in \mathcal{T} (line 2 in Alg. 7) and require that only the subset of \mathcal{T} be used to decrypt this POI. To achieve this, we adopt CP-ABE for encryption. In CP-ABE, a user can decrypt a ciphertext, if and only if their attribute satisfies the policy of the respective ciphertext. For instance, a ciphertext is encrypted with respect to the policy $A \vee B \vee C$; then, the user

Table 6.1. Notations

Symbol	Notation
D	The database of spatial data
m	The number of POIs in D
n	The number of keywords in D
(x, y)	The spatial coordinates
(k, c)	The key-value set composed of keyword and counting
$query = (x, y, \mathcal{Q}_t)$	The spatial query with keyword list
\mathcal{T}	The keyword set of each POI
PK	The public parameters of CP-ABE
MK	The master key of CP-ABE
$EN()$	The encryption of CP-ABE
$\mathcal{A} = (t_1 \vee t_2 \vee \dots \vee t_l)$	The access structure of CP-ABE
I_g	The grid index built by LBSP
$M = m \times m$	The division number of map
EN	The encrypted map
\mathcal{K}	The encryption key generated by LBSP
$\mathcal{N} = \log_2 M$	The number of encryption keys \mathcal{K}
I	The cell ID
\mathcal{K}_I	The encryption for cell I
SK	The secret key of CP-ABE
\mathcal{PQ}	A priority queue
$max()$	The max value of \mathcal{PQ}
$min()$	The min value of \mathcal{PQ}
I	The query index
I_g	The grid index
I_k	The k-quadtrees index
r_i	The random value generated by LBSP
g	The generator of CP-ABE
L	The query result
r'_j	The random value generated by UE
$H()$	The hash function

with a key to attribute A can decrypt this. With the public parameters PK generated according to CP-ABE, the LBSP encrypts each POI with

Algorithm 8: Privacy-Preserving E-kNN - Online Query Processing

Input: LBSP with encryption key, Fog node with encrypted map, UE with $\mathcal{Q} = (x, y, \mathcal{Q}_t)$

Output: UE gets k nearest POIs to \mathcal{Q} with type \mathcal{Q}_t

- 1: UE obtain SK from LBS with \mathcal{Q}_t as input (see Alg. 9)
- 2: UE downloads encrypted map from fog node
- 3: UE inserts all cells of I_g to a priority queue \mathcal{PQ}
- 4: **while** \mathcal{PQ} is not empty and $\max(\mathcal{PQ}) < \min(\mathcal{PQ})$ **do**
- 5: $I = \text{deq}(\mathcal{PQ})$
- 6: LBSP chooses \mathcal{N} random elements r_i and computes $g^{1/\prod_{k=1}^{\mathcal{N}} r_k}$
- 7: LBSP feeds Alg. 10 with input $(K_1^0 r_1, K_1^1 r_1), (K_2^0 r_2, K_2^1 r_2), \dots, (K_{\mathcal{N}}^0 r_{\mathcal{N}}, K_{\mathcal{N}}^1 r_{\mathcal{N}})$
- 8: UE obtains \mathcal{N} keys $K_1^{i_1} r_1, K_2^{i_2} r_2, \dots, K_{\mathcal{N}}^{i_{\mathcal{N}}} r_{\mathcal{N}}$ by feeding I into Alg. 10 also
- 9: UE obtains the decryption key for c_I by $g^{\prod_{k=1}^{\mathcal{N}} K_k^{i_k}} = (g^{1/\prod_{k=1}^{\mathcal{N}} r_k})^{\prod_{k=1}^{\mathcal{N}} K_k^{i_k} r_k}$
- 10: UE obtains encrypted POIs $(EN(m_{x,y,\mathcal{T}}))$ by decrypting $EN(c_I)$, which is, computing $H(g^{\prod_{k=1}^{\mathcal{N}} K_k^{i_k}}, k\alpha) \otimes EN(EN_I)$
- 11: UE decrypts each POI $(EN(m_{x,y,\mathcal{T}}))$ with SK
- 12: UE updates L using newly retrieved POIs
- 13: **end while**
- 14: UE takes POIs in L as the query result

Algorithm 9: Privacy-Preserving Secret Key Generation

Input: LBSP with master key MK , UE with query type

$$\mathcal{Q}_t = t_1, t_2, \dots, t_l$$

Output: UE with SK

- 1: UE generates \mathcal{N} random r_j for each attribute
- 2: LBSP generates a random r and \mathcal{N} random r'_j for each attribute
- 3: LBSP computes $D = g^{(\alpha+r)/\beta}$ and send to UE
- 4: **for** $j = 1$ to \mathcal{N} **do**
- 5: UE computes $H(t_j)^{r_j}$ and sends to LBSP
- 6: LBSP computes $D_j = g^r \cdot (H(t_j)^{r_j})^{r'_j}, D'_j = (g^{r_j})^{r'_j}$
- 7: **end for**
- 8: LBSP sends all D_j and D'_j to UE
- 9: UE constructs $SK = (D, D_j, D'_j) \text{ for } 1 \leq j \leq \mathcal{N}$

the type attributes specified in \mathcal{T} as the access structure. Thus, each POI is encrypted into a unique ciphertext, and only the secret key SK with respect to this access structure can decrypt the POI data correctly. The LBSP creates and encrypts the grid index over the whole region, but each POI is represented as a ciphertext now.

Algorithm 10: Oblivious Key Transfer

Input: LBSP with key generator \mathcal{K} , UE with query index

$$I = i_1, i_2, \dots, i_l$$

Output: UE with unique key $\mathcal{K}_I = \mathcal{K}_1^{i_1}, \mathcal{K}_2^{i_2}, \dots, \mathcal{K}_l^{i_l}$

- 1: **for** $k = 1$ to l **do**
 - 2: with i_k as input, UE uses a 1-out-of-2 OT on $(\mathcal{K}_k^0, \mathcal{K}_k^1)$ to obtain $\mathcal{K}_k^{i_k}$
 - 3: **end for**
-

Online Query Processing

The query described in Alg. 8 follows the best-first strategy. Each time, a priority queue pops up a cell index from the candidate, and with this index and key transfer algorithm (see Alg. 10), the UE can successfully retrieve the corresponding d POIs within the cell. However, it is noteworthy that the message we stored in each cell of I_g is encrypted. So what users obtain with the cell index is d encrypted POIs. To show POIs in plaintext, we only need to get the secret key (SK) with respect to query type \mathcal{Q}_t . The generated SK can decrypt POIs only when \mathcal{Q}_t is a subset of POI type \mathcal{F} .

However, the generation of SK is highly dependent on query type \mathcal{Q}_t and master key MK which are held by the UE and LBSP separately. In addition, SK should only be exposed to the UE at the end. By investigating the detail of CP-ABE, we provide a privacy-preserving secret key generation algorithm (Alg. 9). Instead of sending plaintext of type query \mathcal{Q}_t , we disguise \mathcal{Q}_t with a hash function and random value r_j (line 5 in Alg. 9). SK can be further generated according to this disguised \mathcal{Q}_t and returned to the UE. With this retrieved SK , it is easy for user to decrypt $(EN(m_{x,y}, \mathcal{F}))$ (in line 10 of Alg. 8). We apply the decryption function on each POI in c_i , where only the POI containing query type \mathcal{Q}_t can be decrypted. The whole process repeats until k nearest POIs have been collected.

Security Analysis

The security analysis adopts the *ideal/real simulation paradigm*, which is a standard security formulation in the area of secure multiparty computation [30]. In this formulation, a real protocol execution in a 'real world' is mapped to an ideal protocol realisation in an 'ideal world'. In the ideal world, there exists a secure physical channel, where the adversary can only observe messages that have been sent. A protocol is said to be *secure* if for all adversaries, there exists a simulator so that the outputs of the 'adversary in the real game' and 'simulator in the ideal game' are indistinguishable.

Recall that the E-kNN query is composed of cell key retrieval (oblivious key transfer) and type key retrieval (privacy preserving SK generation). The security of the UE and LBSP during oblivious key transfer is preserved by oblivious transfer. Thus, we only need to prove the security under SK

generation.

Theorem 10. UE's security: *With randomness of r , the E-kNN query hides type \mathcal{Q}_t from the LBSP.*

Proof: We assume that there is a polynomial-time view simulator $Sim_{LBS}(\mathcal{Q}_t)$ for the LBSP server, which generates a distribution statistically close to what is viewed from the LBSP, which is

$$VIEW(\mathcal{Q}_t) = (H(t_1)^{r_1}, H(t_2)^{r_2}, \dots, H(t_i)^{r_i}).$$

For $Sim_{LBSP}(\mathcal{Q}_t)$, it randomly chooses a query \mathcal{Q}_t , which is also denoted as t'_1, t'_2, \dots, t'_i and random $R = (r'_1, r'_2, \dots, r'_i)$. It then computes $H(t'_j)^{r'_j}$ accordingly, during which the view is $VIEW(\mathcal{Q}_t) = (H(t'_1)^{r'_1}, H(t'_2)^{r'_2}, \dots, H(t'_i)^{r'_i})$. This occurs as long as the security of the hash function H and randomness of R , $VIEW(\mathcal{Q}_t)$ and $VIEW(\mathcal{Q}'_t)$ are indistinguishable from each other.

Before we prove the security of the LBSP, we first introduce the DDH assumption that is used as the theoretical foundation for the LBSP security proof.

Theorem 11. DDH assumption: *For a cyclic group G with generator g , given $(g^a, g^b, g^{ab}) \in G$ and $(g^a, g^b, g^c) \in G$, g^{ab} and g^c are computationally indistinguishable.*

Theorem 12. LBSP's security: *Based on the hard problem of the DDH, the E-kNN query prevents the leakage of unauthorised POIs from UE U .*

Proof: Similarly, we define a polynomial-time view simulator $Sim_U(MK)$ for UE U , which generates a distribution statistically close to what is viewed from U , $VIEW(MK) = SK = (D, D_j, D'_j)$. $Sim_U(MK)$ works by randomly generating $MK' = (\beta, g^\alpha)$ and computing D' , D'_j and D''_j according to the keyword (lines 3 and 6 in Alg. 9). The view during this process is $VIEW(MK') = (D', D'_j, D''_j)$. Based on difficulty of the DDH problem, it is easy to prove that $VIEW(MK)$ is computationally indistinguishable from $VIEW(MK')$, which also means the privacy of the LBSP.

6.2.2 Fuzzy kNN Query (F-kNN) Design and Security Analysis

F-kNN allows users to initiate their queries without worrying about their privacy. First, we introduce geo-indistinguishability and text-indistinguishability for the perturbation of location (x, y) and keywords. The notations of this chapter are summarized in Table 6.2.

Geo-indistinguishability

Proposed by Andrés et al. [7], geo-indistinguishability is a formal notion of privacy for location-based systems to protect the user's exact location while allowing an approximate estimation. The definition of geo-indistinguishability is below.

Table 6.2. Notations

Symbol	Notation
(x, y)	The raw coordinates
(x_p, y_p)	The perturbed coordinates
ϵ	The privacy budget
\mathcal{LS}	The acceptable locations set
θ	A random number in $[0, 2\pi)$
v	A random number in $[0, 1)$
$keyword$	The raw keyword
$keyword_p$	The perturbed keyword
\mathcal{KS}	The acceptable keyword set
\mathcal{B}	A binary representation of keywords
M	The POI-keyword matrix
Re	The recommended POI list
Min	The min score of List Re
$sort(,)$	Sorting function
Pos	The position list
$Score$	The overall score of POI
$Score_{sp}$	The spatial score of POI
$Score_{tx}$	The textual score of POI
z	A random integer

Algorithm 11: Implementation of Geo-indistinguishability

Result: perturbed location (x_p, y_p)

Input: raw location (x, y)

Init: Privacy budget ϵ , acceptable locations \mathcal{LS} ;

1: Draw θ in $[0, 2\pi)$;

2: Draw v in $[0, 1)$, set $r \leftarrow C_\epsilon^{-1}(v)$;

3: $z \leftarrow (x, y) + (r\cos(\theta), r\sin(\theta))$;

4: $(x_p, y_p) \leftarrow closet(z, \mathcal{LS})$.

Definition 4 (ϵ -geo-indistinguishability). A randomised mechanism \mathcal{F} satisfies ϵ -geo-indistinguishability iff for any two location inputs $(x, y), (x', y')$ and any output t , it holds

$$Pr(\mathcal{F}(x, y) = t) \leq e^\epsilon \times Pr(\mathcal{F}(x', y') = t).$$

This definition indicates that by observing the output t , the data collector cannot infer whether the input is (x, y) or (x', y') with a confidence higher than e^ϵ . The implementation is shown in Alg. 11. The algorithm draws

Algorithm 12: Implementation of Text-indistinguishability

Result: perturbed keyword $keyword_p$

Input: raw keyword $keyword$

Init: Privacy budget ϵ , acceptable keywords \mathcal{KS} ;

1: Create a vector \mathcal{B} of 26×26 bitlength with each bit corresponding to an alpha pair from ordered list $\{aa, ab, \dots, zz\}$;

2: Set $\mathcal{B}_i = 1$ for $\mathcal{B}_i.label \in k$, otherwise $\mathcal{B}_i = 0$;

3: Perturb \mathcal{B} with

$$\mathcal{B}_i = \begin{cases} \mathcal{B}_i & \text{with probability } p = \frac{e^{\epsilon/26 \times 26}}{1 + e^{\epsilon/26 \times 26}} \\ \bar{\mathcal{B}}_i & \text{with probability } 1 - p \end{cases} \quad (6.1)$$

4: $keyword_p \leftarrow closet(\mathcal{B}, \mathcal{KS})$.

new points by adding a specific Laplace noise decided by value r and θ . θ is a random value in interval $[0, 2\pi)$ with a uniform distribution. r is defined as the reverse of the cumulative distribution function C over ϵ and v as

$$C_\epsilon^{-1}(v) = -\frac{1}{\epsilon} \left(W_{-1} \frac{v-1}{e} + 1 \right),$$

where v is a random value in a uniform distribution $[0, 1)$ and W_{-1} is the Lambert W function, which can be computed efficiently and is implemented in libraries like MATLAB. The detail of geo-indistinguishability can be referred to paper [7].

Text-indistinguishability

Text-indistinguishability is introduced for the privacy protection of keywords. By applying the LDP, it generates a new keyword that meets ϵ -privacy. The formal definition is given below.

Definition 5 (ϵ -text indistinguishability). A randomised algorithm \mathcal{F} satisfies ϵ -text-indistinguishability, iff for any text input $keyword, keyword' \in D$ and for any output t , the following inequality always holds:

$$Pr[\mathcal{F}(keyword) = t] \leq e^\epsilon \times Pr[\mathcal{F}(keyword') = t].$$

Similarly, it preserves ϵ privacy when the data collector cannot infer the input $keyword$ or $keyword'$ from the observed t with a confidence higher than e^ϵ . The implementation is presented in Alg. 12. The keyword is first mapped to a vector according to the algorithm in lines 1 and 2. Here, we adopt the commonly used wording embedding method. We create a vector of 26×26 bit length, and each bit in the vector represents an alpha pair combination. The value is set to '1' when the alpha pair combination is presented in the keyword. For example, 'food' sets '1' on the position {'fo', 'oo' and 'od'}. With the transformed vector, ϵ -LDP can be applied directly by perturbing each bit. However, it is noteworthy that, 26 is selected

Algorithm 13: Improved F-kNN Search

Result: POI recommendation $Re = \{R_1, R_2, \dots, R_k\}$
Input: query Q , k-quadtrees I_k , POI-keyword matrix M
Init: $Min = 1$; Pos ; $Re^q = I_k.knn((x_q, y_q))$

- 1: For each POI in M ;
- 2: Compute $Score_{tx}$ with $\{k_1, k_2, \dots, k_q\}$;
- 3: Sorted POI list $SL \leftarrow sort(M, Score_{tx})$;
- 4: $Re.add(I_k.knn((x_q, y_q)))$;
- 5: For each POI P_i in Re ;
- 6: Compute $Score$;
- 8: $Min = Re.minscore()$;
- 9: $Pos = Max(SL.pos(Re \cap Re^q))$;
- 10: For $i=0, i++$ to Pos ;
- 11: If $i \geq Pos$;
- 12: Break;
- 13: If $P = SL.get(i)$ is not visited;
- 14: Compute $Score(P, Q)$;
- 15: If $Score(P, Q) > Min$;
- 16: $Re.update(P)$;
- 17: $Min = Re.minscore()$;
- 18: $Pos = Max(SL.pos(Re \cap Re^q))$;

Algorithm 14: Budget Distribution Over Unfixed Iterations

Result: infinite series: $\{\epsilon_1, \epsilon_2, \dots, \epsilon_n\}$
Input: privacy budget ϵ

- 1: While True;
- 2: Generate a random integer z ;
- 3: If z is new;
- 4: $\epsilon_{n++} = \frac{\epsilon}{2^{z+1}}$.

according to the number of alpha, and 26×26 is the number of alpha pair combination. Other methods that map the keyword into a 1 – 0 vector can also be applied here. Concretely, the value remains unchanged with probability p and is reversed with probability $1 - p$. The closest keyword to the perturbed vector in the acceptable keyword set \mathcal{KS} is the output.

Fuzzy kNN Query

Based on geo- and text-indistinguishability, a fuzzy kNN query can be implemented. The UE sends a query composed of its location (x, y) and the description list $\{k_1, k_2, \dots, k_q\}$ to the LBSP to obtain POI recommendations. The privacy of the end-user can still be protected with geo-indistinguishability and text-indistinguishability, which results in a perturbed query (x', y') and $\{k'_1, k'_2, \dots, k'_q\}$, respectively. Based on (x', y') , the

LBSP computes $Score_{sp}$ for each POI in database M according to equation 1.4. At the same time, the $Score_{tx}$ for each POI is also computed based on $\{k'_1, k'_2, \dots, k'_q\}$, here according to equation 1.5. By combining $Score_{sp}$ and $Score_{tx}$, the $Score$ for each POI can be obtained. POIs with k highest $Score$ are recommended to the UE.

To reduce the query latency for F-kNN, we have aimed to decrease expensive distance computation operations. One strategy is to narrow down the candidate POIs with Top-k search over location and keywords separately. To achieve this, a quad-tree index I_k and sorted POI list SL are created based on the database. I_k is created by dividing the whole region into four subregions repeatedly until the POIs contained in the subregion are no more than k . The sorted POI list SL is built on keywords. Based on the query keyword, we calculate the textual relevance score $Score_{tx}$ for each POI in database M, as defined in equation 1.5. Then, we sort the POIs topdown with the rank of $Score_{tx}$. The Top-k search is conducted on I_k and SL , as shown in Alg. 13 from lines 4 to 19. Min and Pos are types of information related to vector Re . Min is the minimum POI score in Re , and Pos is the biggest ID number of Re in textual score list SL . Min and Pos are updated whenever Re has executed some operations. Firstly, as shown from lines 4 to 9, the k nearest neighbours are retrieved from k-quadtree I_k and added as candidates into Re . Min and Pos are updated at the same time. From lines 10 to 19, we search all the POIs in SL from 0 to Pos . If there is any POI that has a higher $Score$ than the Top-k candidate POI in Re , we update Re , Min and Pos immediately. The algorithm stops when all the POIs before Pos have been checked, as shown in lines 11 and 12.

Dynamic Distribution of Privacy Budget

This subsection describes the distribution of the privacy budget in F-kNN. Recall that the UE keeps sending their queries to the LBSP. Every UE wants to keep at least ϵ -privacy to its data, which is defined over all its published data records or queries. According to the sequential composition property, the overall privacy is equal to the privacy over a sequence of operations. In other words, this means the overall privacy for each user is the sum of the privacy distributed over every data publication or query. A common practice is to partition ϵ into multiple portions so that each portion can be used by one execution [97]. However, this is feasible only when the number of the portion is known beforehand. Because the number of uploading and querying times for each worker and each user is unpredictable, partitioning is tricky.

We solve this problem by using infinite series. For an infinite series denoted as $\{a_1, a_2, \dots, a_n\}$, the sum S_n of the series converges to a fixed number. An example of a convergent series is $\{1, \frac{1}{2}, \frac{1}{4}, \dots, \frac{1}{2^n}\}$. As n becomes larger, the sum S_n is always lower than 2. Using this theory, we design a privacy budget distribution algorithm for unfixed iterations, as

shown in Alg. 14. For each user of privacy budget ϵ , the algorithm generates a random unrepeatable integer z and applies $\epsilon_i = \frac{\epsilon}{2^{z+1}}$ for the i th iteration. In this way, the overall privacy budget for each user holds with ϵ , and the privacy for the LBS system also holds with ϵ .

Security Analysis

This section analyses the security of the F-kNN query.

Theorem 13. *Given privacy budget ϵ and input (x, y) , the output (x_p, y_p) of Algorithm 11 preserves ϵ indistinguishability.*

Proof. We omit this proof because it can be referred to in [7]. □

Theorem 14. *Given privacy budget ϵ and text input keyword, the output keyword_p of Algorithm 12 preserves ϵ indistinguishability given the perturbation probability*

$$\begin{cases} p = Pr\{1|1, 0|0\} = \frac{e^{\epsilon/26 \times 26}}{1 + e^{\epsilon/26 \times 26}} \\ q = Pr\{1|0, 0|1\} = \frac{1}{1 + e^{\epsilon/26 \times 26}}. \end{cases} \quad (6.2)$$

Proof. Let $keyword, keyword'$ and $\mathcal{B}, \mathcal{B}'$ be any text inputs and the corresponding encoded bit vectors. $keyword_p$ and \mathcal{B}_p are the observing text output and its encoded bit vector generated by algorithm 12. We suppose that there are 26×26 bits in \mathcal{B} . Then, the probability of observing any given text $keyword_p$ by assuming that $keyword$ is known is

$$Pr\{keyword_p|keyword\} = Pr\{\mathcal{B}_p|\mathcal{B}\} = \prod_{i=1}^{26 \times 26} Pr\{\mathcal{B}_p[i]|\mathcal{B}[i]\}.$$

So the probability for the data collector to infer the input in $keyword$ or $keyword'$ can be presented by their ratio as

$$\frac{Pr\{keyword_p|keyword\}}{Pr\{keyword_p|keyword'\}} = \frac{Pr\{\mathcal{B}_p|\mathcal{B}\}}{Pr\{\mathcal{B}_p|\mathcal{B}'\}} = \prod_{i=1}^{26 \times 26} \frac{Pr\{\mathcal{B}_p[i]|\mathcal{B}[i]\}}{Pr\{\mathcal{B}_p[i]|\mathcal{B}'[i]\}}.$$

For simplicity, we start with the ratio computation in one bit. According to the Bayes theorem, we can obtain that for any bit i :

$$\frac{Pr\{\mathcal{B}_p[i]|\mathcal{B}[i]\}}{Pr\{\mathcal{B}_p[i]|\mathcal{B}'[i]\}} = \frac{Pr\{\mathcal{B}[i]|\mathcal{B}_p[i]\}Pr\{\mathcal{B}'[i]\}}{Pr\{\mathcal{B}'[i]|\mathcal{B}_p[i]\}Pr\{\mathcal{B}[i]\}}.$$

It can be easily extended by listing the combination, as follows:

$$\begin{aligned} & \frac{Pr\{0|1\}Pr\{\mathcal{B}'[i]=1\}}{Pr\{1|1\}Pr\{\mathcal{B}[i]=0\}} + \frac{Pr\{1|1\}Pr\{\mathcal{B}'[i]=0\}}{Pr\{0|1\}Pr\{\mathcal{B}[i]=1\}} \\ & + \frac{Pr\{0|0\}Pr\{\mathcal{B}'[i]=1\}}{Pr\{1|0\}Pr\{\mathcal{B}[i]=0\}} + \frac{Pr\{1|0\}Pr\{\mathcal{B}'[i]=0\}}{Pr\{0|0\}Pr\{\mathcal{B}[i]=1\}}. \end{aligned}$$

By substituting this with the perturbation probability defined in Equation 6.2, we have

$$\left(e^{\epsilon/26 \times 26} + \frac{1}{e^{\epsilon/26 \times 26}}\right) \times \left(\frac{Pr\{\mathcal{B}'[i] = 1\}}{Pr\{\mathcal{B}[i] = 0\}} + \frac{Pr\{\mathcal{B}'[i] = 0\}}{Pr\{\mathcal{B}[i] = 1\}}\right).$$

We suppose the bits are randomly distributed in the vector space, thus

$$\frac{Pr\{\mathcal{B}'[i] = 1\}}{Pr\{\mathcal{B}[i] = 0\}} + \frac{Pr\{\mathcal{B}'[i] = 0\}}{Pr\{\mathcal{B}[i] = 1\}} = \frac{1}{2}.$$

Then, we have

$$\frac{Pr\{\mathcal{B}_p[i]|\mathcal{B}[i]\}}{Pr\{\mathcal{B}_p[i]|\mathcal{B}'[i]\}} = \left(e^{\epsilon/26 \times 26} + \frac{1}{e^{\epsilon/26 \times 26}}\right) \times \frac{1}{2}.$$

Because $\epsilon > 0$ and $e^{\epsilon/26 \times 26} > 1$, it is easy to get that

$$\frac{Pr\{\mathcal{B}_p[i]|\mathcal{B}[i]\}}{Pr\{\mathcal{B}_p[i]|\mathcal{B}'[i]\}} < e^{\epsilon/26 \times 26}.$$

Given that there are 26×26 bits in \mathcal{B} , the distinguishability between input *keyword* and *keyword'* is

$$\begin{aligned} \frac{Pr\{keyword_p|keyword\}}{Pr\{keyword_p|keyword'\}} &= \prod_{i=1}^{26 \times 26} \frac{Pr\{\mathcal{B}_p[i]|\mathcal{B}[i]\}}{Pr\{\mathcal{B}_p[i]|\mathcal{B}'[i]\}} \\ &< \prod_{i=1}^{26 \times 26} e^{\epsilon/26 \times 26} = e^\epsilon. \end{aligned}$$

Thus, the theorem is proved. \square

Theorem 15. *The overall privacy of F-kNN is no larger than ϵ when the UE follows the budget distribution in Alg. 14.*

Proof. We assume there are a number of UEs (w) and that each of them sets its privacy budget as $\epsilon_i < \epsilon (1 \leq i \leq w)$.

For UE i with privacy budget ϵ_i , the privacy budget for the q -th query as defined in Alg. 14 is $\epsilon_i^q = \frac{\epsilon_i}{2^{z+1}}$, where z is a random unrepeatable integer. According to the sequential composition theorem, we can conclude that the privacy budget for UE i holds with $\sum_{q=1}^n \epsilon_i^q$, here with n as any integer, which is

$$\sum_{q=1}^n \epsilon_i^q = \sum_{q=1}^n \frac{\epsilon_i}{2^{z+1}} = \frac{\epsilon_i}{2} \sum_{q=1}^n \frac{1}{2^z}.$$

Because $S_n = \sum_{q=1}^n \frac{1}{2^z} < 2$, it is easy to have $\sum_{q=1}^n \epsilon_i^q < \epsilon_i$.

Because the UEs are independent from each other, the overall privacy of F-kNN can be defined as $\max(\epsilon_i) (1 \leq i \leq w)$ according to the parallel composition theorem. We know that for all $1 \leq i \leq w$, $\epsilon_i < \epsilon$, $\max(\epsilon_i) \leq \epsilon$. Herein, the ϵ -privacy for the whole system is preserved. \square

Table 6.3. Summary of the dataset

MAXLON	116.603738	MINLON	116.200006
MAXLAT	40.073334	MINLAT	39.780002
NumType	20	NumPOI	160000

6.3 Performance Evaluation

E-kNN and F-kNN both provide privacy for the LBS query system. E-KNN achieves privacy protection with OT and CP-ABE, and F-kNN achieves privacy with the LDP. This section analyses their performance. All the experiments are performed on a PC with a 2.19 GHz CPU, 4 GB RAM, JDK 8 and Win 10.

6.3.1 Experimental Test of E-kNN query

This analysis discusses query latency in E-kNN and gives a comparison with the state-of-the-art work presented by Yi et al. [107]. Query latency is the time interval from a UE sending a query to receiving feedback, which contains the computing and communication delay.

Experimental Setup

To simplify the comparison resulting from different settings, we unify the problem into a setting where the whole map is divided into $M = m \times m$ cells, with k nearest POIs to the cell center. The performance analysis is based on an average of several retrieval results. In addition, we set the bitlength of all transferred data as 1024. The total type is denoted as $|T|$, and the number of types in the query is denoted as t .

Experimental Dataset

The experiment is taken under a real-world dataset collected in Beijing, China. The information of this dataset is presented in Table 6.3. We crawl the data using Baidu API. The whole dataset is a $33 * 35 km^2$ region with 160,000 POIs.

Experimental Results

In E-kNN, we encrypt each POI with CP-ABE. The key is generated according to keywords. The efficiency comparison between E-kNN and Yi et al. is presented in Fig. 6.1, showing the computation and communication costs recorded, respectively. The comparison is conducted under varying m with $t = 5$, $|T| = 20$. Compared with E-kNN, Yi et al. still has the limitation of scalability. Because the server of Yi et al. needs to iterate over the whole database M , the time cost grows to the second power of m .

Table 6.4. Summary of the Yelp dataset

num. of POI	12,742	num. of user	48,978
num. of Check-in	252,863	num. of keyword	5,161
average keywords per check-ins	7.23		
max keywords for check-ins	145		

Table 6.5. Parameters and settings

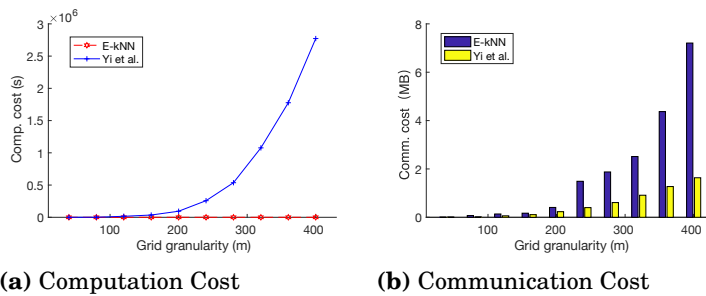
Parameters	Settings
m (number of POI in database)	2,500, 5,000, 7,500, 10,000, 12,500
n (number of keywords in database)	1,000, 2,000, 3,000, 4,000, 5,000
ϵ (privacy budget)	0.2, 0.4, 0.6 , 0.8, 1
q (number of keywords in query)	10, 30, 50 , 70, 90, 110
k (number of recommendation)	5, 10, 15 , 20, 25

6.3.2 Experimental Test of the F-kNN query

This analysis evaluates the accuracy and efficiency performance of F-kNN. The setting of experimental parameters can be referred to in Table 6.5. The default value is given in bold.

Experimental Dataset

An experiment is conducted over a real-world dataset. It is revealed by Yelp in RecSys Challenge 2013, which includes the reviews, POIs, users, and check-ins in Phoenix, Arizona, US. To adapt to our model, the keywords are extracted from the reviews with Porters stemming algorithm in Python [73]. The data statistics after processing are summarised in Table 6.4. A query is generated by combining the randomly selected POIs and keywords. The number of queries is varied according to the requirements, and the number of keywords is set randomly from [1, 10].


Figure 6.1. Efficiency comparison between E-kNN and Yi et al.

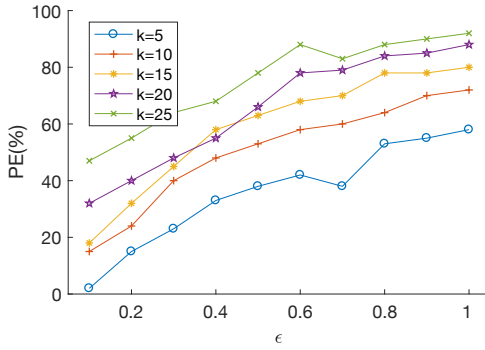


Figure 6.2. Accuracy of F-kNN with variable ϵ

Evaluation Metrics

Precision (PE) is the fraction of relevant POIs among the retrieved POIs, where the retrieved POIs are the ones obtained from raw data without noise and relevant POIs are the ones obtained with LDP protection. The measurement efficiency focuses on the computation and communication costs, which are reflected by the computation time and size of the communication data.

Experimental Results of Accuracy and Efficiency

Accuracy PE is used to measure the accuracy of Top- k recommendations after applying the LDP. Fig. 6.2 shows the recommendation precision with a varying privacy budget ϵ of different recommendation number k . The result shows that PE grows when ϵ increases. Similarly, this is because a smaller ϵ introduces more noise, which decreases the recommendation accuracy. The precision converges while ϵ is close to 1. In addition, we can observe that for a fixed ϵ , the precision grows significantly (around 50%), while k increases from 5 to 25. According to the property of the LDP, we know that although the POI scores are biased by LDP noise, the scores still follow the statistic distribution of the original ones. It is easier to hit the right POIs when k is large. Thus, it is recommended to increase k to obtain a better recommendation accuracy.

Efficiency Efficiency is compared between the normal kNN algorithm (Co-Topk) and our improved kNN algorithm (Qu-Topk). A query is composed of a random location and q random keywords. The performance is shown with the average results over 100 rounds. Fig. 6.3 shows the performance with the variables m , n and q , respectively. From Fig. 6.3(a), (c) and (e), we can see that both the computation time of Co-Topk and Qu-Topk increase linearly with an increase of m , but it stays stable with an increase of n and q . This is because the distance computation incurred by increasing m is much heavier than the keyword computation incurred by n or q . In general, the computation time of Qu-Topk is much smaller than Co-

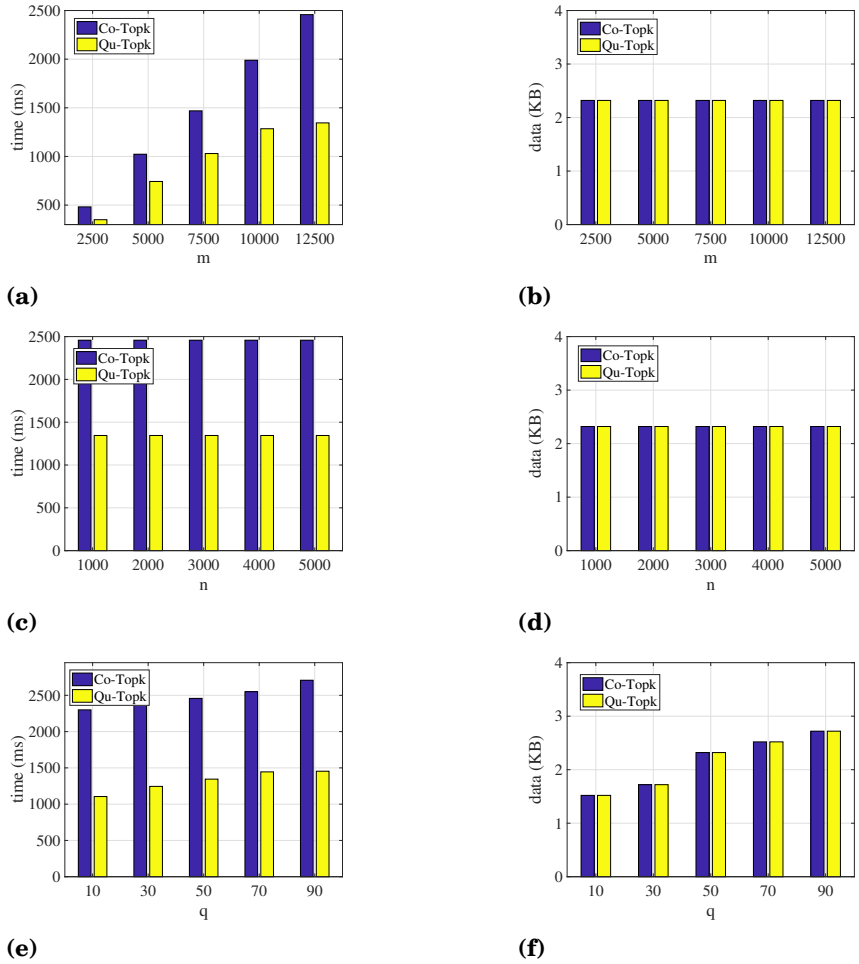


Figure 6.3. Efficiency comparison of F-kNN with the variables m , n and q

Topk. This is because of the efficient pruning strategy of Qu-Topk. For the communication cost, Co-Topk remains the same as Qu-Topk, as shown in Fig. 6.3(b), (d) and (f).

6.4 Summary

This chapter proposed two low-latency LBS query schemes (E-kNN and F-kNN) to support mobile queries in the 5G network. Specifically, E-kNN is implemented with the ability to support privacy protection for accurate kNN queries. In contrast, F-kNN achieves better performance than E-kNN but at the compromise of accuracy, which only supports approximate kNN query. In practice, the scheme selection is highly dependent on the sensitivity of results accuracy. Besides, the proposed schemes still have

some limitations in practical deployment since it may not be that easy to realize full coverage of the whole map considering the overhead of storage and communication, and uncovered regions may occur sometimes.

7. Conclusion and Future Perspectives

This chapter concludes the dissertation with a discussion of the current work and areas for future research.

7.1 Conclusion

Focusing on 5G positioning and its services, the current dissertation examined the security and privacy protection of 5G positioning, verifiable positioning in an edge computing environment, privacy protection of D2D cooperative location verification and privacy-preserving LBS queries. For each research problem, we proposed, improved and carefully evaluated solutions using solid experiments. In summary, the contributions of the present dissertation are as follows:

- To mitigate the influence of adversary attacks (radio jamming attack, and collusion attack) in the process of 5G positioning, we proposed a clustering-based truth discovery scheme that can preserve high positioning accuracy when facing these attacks. Additionally, two neural network models were trained to classify potential attacks and trace attack sources to remove obstacles. In addition, two protocols with different privacy constraints (Pub-pos and Pri-pos) were proposed to mitigate the privacy risk in the emerging outsourced positioning computation environment. The designed protocols were found to be useful in many use cases. For example, Pub-pos can be helpful in privacy preservation for outdoor positioning that is based on base stations [13] or VANET positioning that is based on roadside units [92], where the locations of reference infrastructures are normally known. Pri-pos can be applied to scenarios where not only the distance, but also the coordinates of references, are quite sensitive and request protection, for example, indoor positioning based on crowdsourcing workers or VANET positioning based on vehicles.
- To ensure the integrity of positioning services that are outsourced

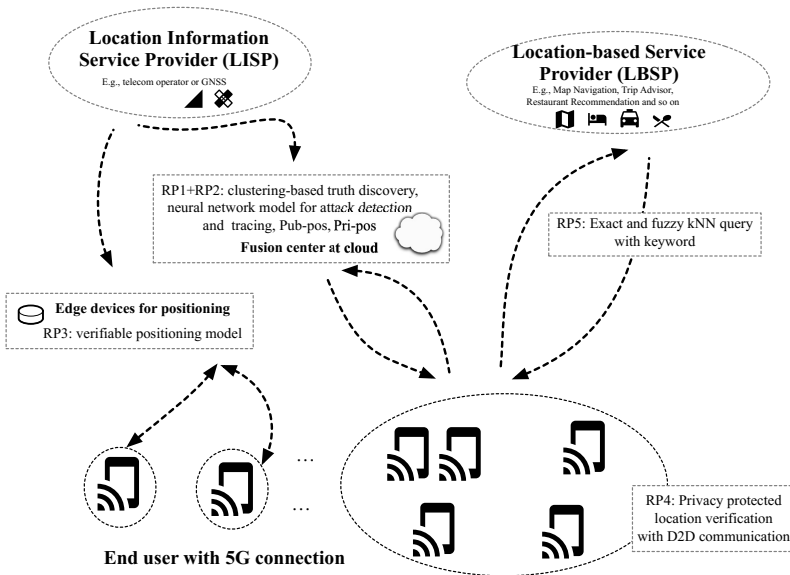


Figure 7.1. The integration of solutions within 5G positioning ecosystem

at edge points, a verifiable positioning model based on the 'backdoor' concept was provided. The new model supported effective verification while keeping high positioning accuracy. Also, to remove the participants' concern about location exposure during D2D cooperative location verification, an efficient scheme based on OPE was presented and implemented. The proposed solution was found to outperform traditional solutions in the balance of privacy, utility and performance.

- To protect the data privacy of both the user and LBSP, support multiple LBS queries and meet the low-latency requirement from LBS services, the current dissertation proposed two privacy-preserving schemes (exact and fuzzy) based on the kNN query. The proposed schemes can be easily adapted to other LBS queries by changing the setting of k . The experiments showed that the proposed schemes outperformed other solutions, having a great advantage in online latency, which is crucial in 5G positioning scenarios.

7.2 Applicability and Limitations

The present dissertation has contributed solutions to the main research problems. The integration of solutions with the 5G positioning ecosystem is demonstrated in Fig 7.1. A further discussion of the applicability and limitations of the proposed solutions are as follows:

For the research of security and privacy protection in 5G positioning (RP1 and RP2), we proposed three modules and two protocols. The practical implementation of these modules and protocols can be conducted by the position service provider, which here would normally be a Telecom operator. The modules and protocols can be integrated into the operator's fusion center and put into service whenever a positioning request is generated. Because they are just algorithms with signal data as the inputs, they can be deployed directly. These solutions are not only effective in 5G positioning, but they can also be applied in more general cases. For example, clustering-based truth discovery and model based attack detection and tracing are useful for areas including positioning in the industrial internet of things, unmanned aerial vehicles and so on. Pub-pos and Pri-pos can be adapted to the outsourcing verification of linear equations. Although these solutions are effective in theory, their applicability, in reality, is still challenging. The threshold of truth discovery and feature selection of neural network models are context specific, which means they need to be updated constantly. Also, the selection of these parameters is heavily dependent on the historical data. For a situation when the region is newly built and there are no historical data available, we can train the model with simulated data and keep the model updated once new data become available. However, this also comes with the possible misjudging of attack detection. Pub-pos and Pri-pos are not dependent on data; that being said, the latency from extra protection computations should be further decreased.

For the design of verifiable positioning models in outsourcing (RP3), we integrated a backdoor into model training. The implementation of this method can be conducted by the LISP during positioning model training. The solution can also be immigrated into a general case of outsourced model verification with an adaption of the trigger dataset. For example, this can be applied in the integrity check of outsourced image recognition. The limitation of the proposed solution is that it only allows the owner model to conduct the integrity check. The model users are unable to conduct the integrity check, limiting its usage scale. Second, we contributed to privacy-protected D2D cooperative location verification (RP4), which allows the users to verify their locations with surrounding devices without revealing any sensitive information. For any two users who want privacy regarding their location verification, they can call the protocol API directly without extra effort. It is easy to deploy and use in practice. However, the extra communicational and computational costs could cause a long delay for the end-users and consume additional energy on the phone.

The privacy-preserving LBS shows high efficiency (RP5), but it is also remarkable that the proposed E-kNN and F-kNN queries still have some limitations. It may not be that easy to apply them in the whole map since the high overhead, and uncovered regions may occur sometimes. The query flexibility still needs to be improved to offer a better user experience. For

example, a query can be like the nearest restaurant that opens in the afternoon.

7.3 Future Perspectives

The privacy, security and trust management of positioning is highly dependent on foundational communication technologies. However, these solutions are not adequate for upcoming new communication technologies like 6G. The present dissertation concludes by highlighting foundational research challenges, as well as implications and opportunities related to privacy, security and trust in the future.

(1) Positioning attack removal and detection for ground, sky and sea: a future positioning service should aim to achieve coverage on the ground, sky and sea. As the main solution for maintaining positioning accuracy, novel attack detection and mitigation technologies are expected to overcome the challenges in these environments, especially for services out on sea. Compared with other environments, positioning on the sea is the most challenging because the influence of water currents and oceanographic animals also complicates attack detection and removal. Positioning services leveraging AI techniques are essential to autonomously identify and respond to potential threats based on anomalies.

(2) Security, privacy and trust of AI-enabled positioning model: AI-enabled positioning has been widely studied as a way to provide positioning services with Wi-Fi signals, channel state information (CSI) and so forth. Future networks are also expected to be equipped with ubiquitous AI services. However, their vulnerability to adversarial machine learning attacks, data poisoning attacks, transfer learning attacks and data phishing attacks still exists and needs to be solved. The privacy risks because of the openness of the network and edge/fog outsourcing also raise concerns. Model trust needs to be enhanced using effective verification. Although the present dissertation proposed solutions to solve the security, privacy and trust issue of ML models outsourced to cloud/edge nodes, we still look forward to light, efficient solutions that can be applied in more resource-limited networks such as in-body networks and environmental sensor networks.

(3) Positioning security and privacy protection at the physical layer and against quantum computer-based attacks: with emerging communication technologies, positioning and sensing are expected to coexist with communication, making the defence on data and application level invalid. Thus, the research should place greater emphasis on the physical layer's security design. Also, quantum computation is expected since privacy protection solutions based on traditional NP-hard problems are not hard enough under quantum computer-based attacks.

References

- [1] Aly Sabri Abdalla, Keith Powell, Vuk Marojevic, and Giovanni Geraci. UAV-assisted attack prevention, detection, and recovery of 5G networks. *IEEE Wireless Communications*, 27(4):40–47, 2020.
- [2] Abbas Acar, Hidayet Aksu, A Selcuk Uluagac, and Mauro Conti. A survey on homomorphic encryption schemes: Theory and implementation. *ACM Computing Surveys (Csur)*, 51(4):1–35, 2018.
- [3] Yossi Adi, Carsten Baum, Moustapha Cisse, Benny Pinkas, and Joseph Keshet. Turning your weakness into a strength: Watermarking deep neural networks by backdooring. In *27th {USENIX} Security Symposium ({USENIX} Security 18)*, pages 1615–1631, 2018.
- [4] Rakesh Agrawal, Jerry Kiernan, Ramakrishnan Srikant, and Yirong Xu. Order preserving encryption for numeric data. In *Proceedings of the 2004 ACM SIGMOD international conference on Management of data*, pages 563–574, 2004.
- [5] Raed S Alharthi, Esam Aloufi, Ibrahim Alrashdi, Ali Alqazzaz, Mohamed A Zohdy, and Julian L Rrushi. Protecting location privacy for crowd workers in spatial crowdsourcing using a novel dummy-based mechanism. *IEEE Access*, 8:114608–114622, 2020.
- [6] NGMN Alliance. 5G white paper. *Next generation mobile networks, white paper*, 1(2015), 2015.
- [7] Miguel E Andrés, Nicolás E Bordenabe, Konstantinos Chatzikokolakis, and Catuscia Palamidessi. Geo-indistinguishability: Differential privacy for location-based systems. In *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, pages 901–914, 2013.
- [8] Sanjeev Arora and Shmuel Safra. Probabilistic checking of proofs: A new characterization of NP. *Journal of the ACM (JACM)*, 45(1):70–122, 1998.
- [9] László Babai, Lance Fortnow, Leonid A Levin, and Mario Szegedy. Checking computations in polylogarithmic time. In *Proceedings of the twenty-third annual ACM symposium on Theory of computing*, pages 21–32, 1991.
- [10] Bhuvan Bamba, Ling Liu, Peter Pesti, and Ting Wang. Supporting anonymous location queries in mobile environments with privacygrid. In *International Conference on World Wide Web*, pages 237–246, 2008.
- [11] John Bethencourt, Amit Sahai, and Brent Waters. Ciphertext-policy attribute-based encryption. In *Security and Privacy, 2007. SP'07. IEEE Symposium on*, pages 321–334. IEEE, 2007.

- [12] Claudio Bettini, X. Sean Wang, and Sushil Jajodia. Protecting privacy against location-based personal identification. In *Vldb International Conference on Secure Data Management*, pages 185–199, 2005.
- [13] Andrej Bogdanov, Elitza Maneva, and Samantha Riesenfeld. Power-aware base station positioning for sensor networks. In *IEEE INFOCOM 2004*, volume 1. IEEE, 2004.
- [14] Jin Cao, Zheng Yan, Ruhui Ma, Yinghui Zhang, Yulong Fu, and Hui Li. LSAA: a lightweight and secure access authentication scheme for both UE and mMTC devices in 5G networks. *IEEE Internet of Things Journal*, 7(6):5329–5344, 2020.
- [15] Srdjan Capkun and Jean-Pierre Hubaux. Secure positioning of wireless devices with application to sensor networks. In *Proceedings IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 3, pages 1917–1928. IEEE, 2005.
- [16] Jing Chen, Kun He, Quan Yuan, Min Chen, Ruiying Du, and Yang Xiang. Blind filtering at third parties: An efficient privacy-preserving framework for location-based services. *IEEE Transactions on Mobile Computing*, 17(11):2524–2535, 2018.
- [17] Chi-Yin Chow, Mohamed F Mokbel, and Xuan Liu. A peer-to-peer spatial cloaking algorithm for anonymous location-based service. In *Proceedings of the 14th annual ACM international symposium on Advances in geographic information systems*, pages 171–178, 2006.
- [18] Moustapha M Cisse, Yossi Adi, Natalia Neverova, and Joseph Keshet. Houdini: Fooling deep structured visual and speech recognition models with adversarial examples. In *Advances in neural information processing systems*, pages 6977–6987, 2017.
- [19] Armin Dammann, Ronald Raulefs, and Siwei Zhang. On prospects of positioning in 5G. In *2015 IEEE International Conference on Communication Workshop (ICCW)*, pages 1207–1213. IEEE, 2015.
- [20] JA del Peral-Rosado, GS Granados, R Raulefs, E Leitinger, S Grebien, T Wilding, D Dardari, ES Lohan, H Wymeersch, JJ Floch, et al. Whitepaper on new localization methods for 5G wireless systems and the Internet-of-Things. In *White Paper of the COST Action CA15104 (IRACON)*, pages 1–27. COST Action CA15104, IRACON, 2018.
- [21] Ivan Dokmanic, Reza Parhizkar, Juri Ranieri, and Martin Vetterli. Euclidean distance matrices: essential theory, algorithms, and applications. *IEEE Signal Processing Magazine*, 32(6):12–30, 2015.
- [22] Jia Duan, Jiantao Zhou, and Yuanman Li. Secure and verifiable outsourcing of nonnegative matrix factorization (NMF). In *Proceedings of the 4th ACM Workshop on Information Hiding and Multimedia Security*, pages 63–68. ACM, 2016.
- [23] Buğra Gedik and Ling Liu. Location privacy in mobile systems: A personalized anonymization model. In *IEEE International Conference on Distributed Computing Systems, 2005. ICDCS 2005. Proceedings*, pages 620–629, 2005.
- [24] Gabriel Ghinita, Panos Kalnis, Ali Khoshgozaran, Cyrus Shahabi, and Kian-Lee Tan. Private queries in location based services: anonymizers are not necessary. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 121–132, 2008.

- [25] Zahra Ghodsi, Tianyu Gu, and Siddharth Garg. Safetynets: Verifiable execution of deep neural networks on an untrusted cloud. *Advances in Neural Information Processing Systems*, 30, 2017.
- [26] Shafi Goldwasser, Silvio Micali, and Charles Rackoff. The knowledge complexity of interactive proof systems. *SIAM Journal on computing*, 18(1):186–208, 1989.
- [27] Manish Gupta, Jing Gao, Charu Aggarwal, and Jiawei Han. Outlier detection for temporal data. *Synthesis Lectures on Data Mining and Knowledge Discovery*, 5(1):1–129, 2014.
- [28] Ismail Guvenc and Chia-Chin Chong. A survey on TOA based wireless localization and NLOS mitigation techniques. *IEEE Communications Surveys & Tutorials*, 11(3):107–124, 2009.
- [29] Zhanjun Hao, Yan Yan, Xiaochao Dang, and Chenguang Shao. Endpoints-clipping CSI amplitude for SVM-based indoor localization. *Sensors*, 19(17):3689, 2019.
- [30] Carmit Hazay and Yehuda Lindell. *Efficient secure two-party protocols: Techniques and constructions*. Springer Science Business Media, 2010.
- [31] Changqin Huang, Ming Ma, Yuxin Liu, and Anfeng Liu. Preserving source location privacy for energy harvesting WSNs. *Sensors*, 17(4):724, 2017.
- [32] Yan Huang, David Evans, Jonathan Katz, and Lior Malka. Faster secure two-party computation using garbled circuits. In *20th USENIX Security Symposium (USENIX Security 11)*, 2011.
- [33] Siam Umar Hussain and Farinaz Koushanfar. P3: Privacy preserving positioning for smart automotive systems. *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, 23(6):1–19, 2018.
- [34] Mai Ibrahim, Marwan Torki, and Mustafa Elnainay. CNN based indoor localization using RSS time-series. In *2018 IEEE Symposium on Computers and Communications (ISCC)*, pages 01044–01049. IEEE, 2018.
- [35] Kai Jansen, Matthias Schäfer, Daniel Moser, Vincent Lenders, Christina Pöpper, and Jens Schmitt. Crowd-GPS-Sec: Leveraging crowdsourcing to detect and localize GPS spoofing attacks. In *IEEE Symposium on Security and Privacy (SP)*, pages 1018–1031. IEEE, 2018.
- [36] Han Jiang, Hao Wang, Zhihua Zheng, and Qiuliang Xu. Privacy preserved wireless sensor location protocols based on mobile edge computing. *Computers & Security*, 84:393–401, 2019.
- [37] Jiawen Kang, Rong Yu, Xumin Huang, Magnus Jonsson, Hanna Bogucka, Stein Gjessing, and Yan Zhang. Location privacy attacks and defenses in cloud-enabled Internet of Vehicles. *IEEE Wireless Communications*, 23(5):52–59, 2016.
- [38] Florian Kerschbaum. Commutative order-preserving encryption, May 17 2012. US Patent App. 12/944,672.
- [39] Syed Khandker, Joaquín Torres-Sospedra, and Tapani Ristaniemi. Improving RF fingerprinting methods by means of D2D communication protocol. *Electronics*, 8(1):97, 2019.
- [40] H. Kido, Y. Yanagisawa, and T. Satoh. An anonymous communication technique using dummies for location-based services. In *International Conference on Pervasive Services*, pages 88–97, 2005.

- [41] Jong Seon Kim, Jong Wook Kim, and Yon Dohn Chung. Successive point-of-interest recommendation with local differential privacy. *IEEE Access*, 9:66371–66386, 2021.
- [42] Steven L Kinney. *Trusted platform module basics: using TPM in embedded systems*. Elsevier, 2006.
- [43] Mike Koivisto, Aki Hakkarainen, Mário Costa, Jukka Talvitie, Kari Heiska, Kari Leppänen, and Mikko Valkama. Continuous high-accuracy radio positioning of cars in ultra-dense 5G networks. In *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC)*, pages 115–120. IEEE, 2017.
- [44] Vladimir Kolesnikov and Thomas Schneider. Improved garbled circuit: Free xor gates and applications. In *International Colloquium on Automata, Languages, and Programming*, pages 486–498. Springer, 2008.
- [45] Felix Kreuk, Yossi Adi, Moustapha Cisse, and Joseph Keshet. Fooling end-to-end speaker verification with adversarial examples. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1962–1966. IEEE, 2018.
- [46] Li Kuang, Shuai He, Yuyou Fan, Huan Zhang, and Ruyi Shi. T-SR: A location privacy protection algorithm based on POI query. *IEEE Access*, 7:59491–59503, 2019.
- [47] Hong Li, Limin Sun, Haojin Zhu, Xiang Lu, and Xiuzhen Cheng. Achieving privacy preservation in WiFi fingerprint-based localization. In *IEEE Infocom Conference on Computer Communications*, pages 2337–2345. IEEE, 2014.
- [48] Hongtao Li, Xingsi Xue, Zhiying Li, Long Li, and Jinbo Xiong. Location privacy protection scheme for LBS in IoT. *Wireless Communications and Mobile Computing*, 2021, 2021.
- [49] Bozhong Liu, Ling Chen, Xingquan Zhu, Ying Zhang, Chengqi Zhang, and Weidong Qiu. Protecting location privacy in spatial crowdsourcing using encrypted data. *Advances in Database Technology-EDBT*, 2017.
- [50] Hai Liu, Xinghua Li, Hui Li, Jianfeng Ma, and Xindi Ma. Spatiotemporal correlation-aware dummy-based privacy protection scheme for location-based services. In *IEEE International Conference on Computer Communications*, pages 1–9. IEEE, 2017.
- [51] Shushu Liu, An Liu, Lei Zhao, Guanfeng Liu, Zhixu Li, Pengpeng Zhao, Kai Zheng, and Lu Qin. Efficient query processing with mutual privacy protection for location-based services. In *International Conference on Database Systems for Advanced Applications*, pages 299–313. Springer, 2016.
- [52] Xuefeng Liu, Wenhai Sun, Hanyu Quan, Wenjing Lou, Yuqing Zhang, and Hui Li. Publicly verifiable inner product evaluation over outsourced data streams under multiple keys. *IEEE Transactions on Services Computing*, 10(5):826–838, 2016.
- [53] Elena Simona Lohan, Anette Alén-Savikko, Liang Chen, Kimmo Järvinen, Helena Leppäkoski, Heidi Kuusniemi, and Päivi Korpisaari. 5G positioning: Security and privacy aspects. *A Comprehensive Guide to 5G Security*, pages 281–320, 2018.
- [54] Elena Simona Lohan, Jukka Talvitie, Pedro Figueiredo e Silva, Henri Nurminen, Simo Ali-Löytty, and Robert Piché. Received signal strength models for WLAN and BLE-based indoor positioning in multi-floor buildings. In *International Conference on Localization and GNSS*, pages 1–6. IEEE, 2015.

- [55] Lung Yiu Man, Christian S. Jensen, Xuegang Huang, and Hua Lu. SpaceTwist: Managing the trade-offs among location privacy, query performance, and query accuracy in mobile services. In *IEEE International Conference on Data Engineering*, pages 366–375, 2008.
- [56] Mohsen Riahi Manesh, Jonathan Kenney, Wen Chen Hu, Vijaya Kumar Devabhaktuni, and Naima Kaabouch. Detection of GPS spoofing attacks on unmanned aerial systems. In *2019 16th IEEE Annual Consumer Communications & Networking Conference (CCNC)*, pages 1–6. IEEE, 2019.
- [57] Charalampos Mavroforakis, Nathan Chenette, Adam O’Neill, George Kollios, and Ran Canetti. Modular order-preserving encryption, revisited. In *International Conference on Management of Data (SIGMOD)*, pages 763–777, 2015.
- [58] Rico Mendrzik, Henk Wymeersch, Gerhard Bauch, and Zohair Abu-Shaban. Harnessing NLOS components for position and orientation estimation in 5G millimeter wave MIMO. *IEEE Transactions on Wireless Communications*, 18(1):93–107, 2018.
- [59] Estifanos Yohannes Menta, Nicolas Malm, Riku Jäntti, Kalle Ruttik, Mário Costa, and Kari Leppänen. On the performance of AoA-based localization in 5G ultra-dense networks. *IEEE Access*, 7:33870–33880, 2019.
- [60] W Mohr. 5G empowering vertical industries. In *Tech. Rep. 5G PPP*, 2016.
- [61] Mohamed F Mokbel, Chi Yin Chow, and Walid G Aref. The new casper: query processing for location services without compromising privacy. In *International Conference on Very Large Data Bases*, pages 763–774, 2006.
- [62] Moni Naor and Benny Pinkas. Oblivious transfer with adaptive queries. *Lecture Notes in Computer Science*, 1666:573–590, 1999.
- [63] Arvind Narayanan, Narendran Thiagarajan, Mugdha Lakhani, Michael Hamburg, Dan Boneh, et al. Location privacy via private proximity testing. In *Network and Distributed System Security Symposium (NDSS)*, volume 11, 2011.
- [64] Ben Niu, Qinghua Li, Xiaoyan Zhu, Guohong Cao, and Hui Li. Achieving k-anonymity in privacy-aware location-based services. In *IEEE Conference on Computer Communications (INFOCOM)*, pages 754–762. IEEE, 2014.
- [65] Mohammad Reza Nosouhi, Vu Viet Hoang Pham, Shui Yu, Yong Xiang, and Matthew Warren. A hybrid location privacy protection scheme in big data environment. In *IEEE Global Communications Conference (GLOBECOM)*, pages 1–6. IEEE, 2017.
- [66] Vuokko Nurmela et al. Deliverable D1. 4 METIS channel models. In *Proc. Mobile Wireless Commun. Enablers Inf. Soc. (METIS)*, page 1, 2015.
- [67] Jongtaek Oh and Jisu Kim. Adaptive k-nearest neighbour algorithm for WiFi fingerprint positioning. *Ict Express*, 4(2):91–94, 2018.
- [68] Opeyemi Osanaiye, Shuo Chen, Zheng Yan, Rongxing Lu, Kim-Kwang Raymond Choo, and Mqhele Dlodlo. From cloud to fog computing: A review and a conceptual live VM migration framework. *IEEE Access*, 5:8284–8300, 2017.
- [69] Pascal Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *International conference on the theory and applications of cryptographic techniques*, pages 223–238. Springer, 1999.

- [70] Russell Paulet, Md Golam Kaosar, Xun Yi, and Elisa Bertino. Privacy-preserving and content-protecting location based queries. *IEEE transactions on knowledge and data engineering*, 26(5):1200–1210, 2013.
- [71] Nicholas Peccarelli, Blake James, Robin Irazoqui, Justin Metcalf, Caleb Fulton, and Mark Yeary. Survey: Characterization and mitigation of spatial/spectral interferers and transceiver nonlinearities for 5G MIMO systems. *IEEE Transactions on Microwave Theory and Techniques*, 67(7):2829–2846, 2019.
- [72] Qianwen Pei, Burong Kang, Lei Zhang, Kim-Kwang Raymond Choo, Yuanfei Zhang, and Yinxia Sun. Secure and privacy-preserving 3D vehicle positioning schemes for vehicular ad hoc network. *EURASIP Journal on Wireless Communications and Networking*, 2018(1):1–12, 2018.
- [73] Martin F Porter. An algorithm for suffix stripping. *Program*, 2006.
- [74] Yihong Qi, Hisashi Kobayashi, and Hirohito Suda. Analysis of wireless geolocation in a non-line-of-sight environment. *IEEE Transactions on wireless communications*, 5(3):672–681, 2006.
- [75] Amir Mahdi Sazdar, Nasim Alikhani, Seyed Ali Ghorashi, and Ahmad Khonsari. Privacy preserving in indoor fingerprint localization and radio map expansion. *Peer-to-Peer Networking and Applications*, 14(1):121–134, 2021.
- [76] Jochen Schiller and Agnès Voisard. *Location-based services*. Elsevier, 2004.
- [77] Erich Schubert, Jörg Sander, Martin Ester, Hans Peter Kriegel, and Xiaowei Xu. DBSCAN revisited: why and how you should (still) use DBSCAN. *ACM Transactions on Database Systems (TODS)*, 42(3):1–21, 2017.
- [78] Felix Schuster, Manuel Costa, Cédric Fournet, Christos Gkantsidis, Marcus Peinado, Gloria Mainar-Ruiz, and Mark Russinovich. VC3: Trustworthy data analytics in the cloud using SGX. In *2015 IEEE Symposium on Security and Privacy*, pages 38–54. IEEE, 2015.
- [79] Matthias Schäfer, Carolina Nogueira, Jens Schmitt, and Vincent Lenders. Secure location verification: Why you want your verifiers to be mobile. In *International Workshop on Attacks and Defenses for Internet-of-Things (ADIoT) 2019, co-located with ESORICS*, September 2019.
- [80] Arash Shahmansoori, Gabriel E Garcia, Giuseppe Destino, Gonzalo Seco-Granados, and Henk Wymeersch. 5G position and orientation estimation through millimeter wave MIMO. In *2015 IEEE Globecom Workshops (GC Wkshps)*, pages 1–6. IEEE, 2015.
- [81] Jun Shao, Rongxing Lu, and Xiaodong Lin. FINE: A fine-grained privacy-preserving location-based service framework for mobile devices. In *IEEE conference on computer communications (INFOCOM)*, pages 244–252. IEEE, 2014.
- [82] Vishal Sharma, Ilsun You, and Nadra Guizani. Security of 5G-V2X: Technologies, standardization, and research directions. *IEEE Network*, 34(5):306–314, 2020.
- [83] Weisong Shi, Jie Cao, Quan Zhang, Youhuizi Li, and Lanyu Xu. Edge computing: Vision and challenges. *IEEE Internet of Things Journal*, 3(5):637–646, 2016.
- [84] Kang G Shin, Xiaoen Ju, Zhigang Chen, and Xin Hu. Privacy protection for users of location-based services. *IEEE Wireless Communications*, 19(1):30–39, 2012.

- [85] Tao Shu, Yingying Chen, Jie Yang, and Albert Williams. Multi-lateral privacy-preserving localization in pervasive environments. In *IEEE conference on computer communications (INFOCOM)*, pages 2319–2327. IEEE, 2014.
- [86] Gunnar A Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *European Conference on Computer Vision*, pages 510–526. Springer, 2016.
- [87] Mridula Singh, Patrick Leu, AbdelRahman Abdou, and Srdjan Capkun. UWB-ED: distance enlargement attack detection in ultra-wideband. In *28th {USENIX} Security Symposium Security 19*, pages 73–88, 2019.
- [88] Sean W Smith and Steve Weingart. Building a high-performance, programmable secure coprocessor. *Computer Networks*, 31(8):831–860, 1999.
- [89] Pablo Speciale, Johannes L Schonberger, Sing Bing Kang, Sudipta N Sinha, and Marc Pollefeys. Privacy preserving image-based localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5493–5503, 2019.
- [90] Hien To, Gabriel Ghinita, and Cyrus Shahabi. A framework for protecting worker location privacy in spatial crowdsourcing. *Proceedings of the VLDB Endowment*, 7(10):919–930, 2014.
- [91] Joaquín Torres-Sospedra, Raúl Montoliu, Adolfo Martínez-Usó, Joan P Avariento, Tomás J Arnau, Mauri Benedito-Bordonau, and Joaquín Huerta. UJIIndoorLoc: A new multi-building and multi-floor database for WLANfingerprint-based indoor localization problems. In *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 261–270. IEEE, 2014.
- [92] Ming-Fong Tsai, Po-Ching Wang, Ce-Kuen Shieh, Wen-Shyang Hwang, Naveen Chilamkurti, Seungmin Rho, and Yang Sun Lee. Improving positioning accuracy for VANET in real city environments. *The Journal of Supercomputing*, 71(6):1975–1995, 2015.
- [93] Michael Walfish and Andrew J Blumberg. Verifying computations without reexecuting them. *Communications of the ACM*, 58(2):74–84, 2015.
- [94] Lingling Wang, Guozhu Liu, and Lijun Sun. A secure and privacy-preserving navigation scheme using spatial crowdsourcing in fog-based VANETs. *Sensors*, 17(4):668, 2017.
- [95] Qian Wang, Zhaojun Lu, Mingze Gao, and Gang Qu. Edge computing based GPS spoofing detection methods. In *IEEE 23rd International Conference on Digital Signal Processing (DSP)*, pages 1–5. IEEE, 2018.
- [96] Weiqi Wang, An Liu, Zhixu Li, Xiangliang Zhang, Qing Li, and Xiaofang Zhou. Protecting multi-party privacy in location-aware social point-of-interest recommendation. *World Wide Web*, 22(2):863–883, 2019.
- [97] Jianhao Wei, Yaping Lin, Xin Yao, and Jin Zhang. Differential privacy-based location protection in spatial crowdsourcing. *IEEE Transactions on Services Computing*, 2019.
- [98] Hongfeng Wu and Jingjing Yan. Outsourcing computing of large matrix Jordan decomposition. *Mathematical Problems in Engineering*, 2019, 2019.
- [99] Henk Wymeersch, Gonzalo Seco-Granados, Giuseppe Destino, Davide Dardari, and Fredrik Tufvesson. 5G mmWave positioning for vehicular networks. *IEEE Wireless Communications*, 24(6):80–86, 2017.

- [100] Linchen Xiao, Arash Behboodi, and Rudolf Mathar. A deep learning approach to fingerprinting indoor localization solutions. In *27th International Telecommunication Networks and Applications Conference (ITNAC)*, pages 1–7. IEEE, 2017.
- [101] Yonghui Xiao and Li Xiong. Protecting locations with differential privacy under temporal correlations. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pages 1298–1309, 2015.
- [102] Guowen Xu, Hongwei Li, Yuanshun Dai, Kan Yang, and Xiaodong Lin. Enabling efficient and geometric range query with access control over encrypted spatial data. *IEEE Transactions on Information Forensics and Security*, 14(4):870–885, 2018.
- [103] Zheng Yan, Xixun Yu, and Wenxiu Ding. Context-aware verifiable cloud computing. *IEEE Access*, 5:2211–2227, 2017.
- [104] Xue Yang, Fan Yin, and Xiaohu Tang. A fine-grained and privacy-preserving query scheme for fog computing-enhanced location-based service. *Sensors*, 17(7):1611, 2017.
- [105] Zheng Yang and Kimmo Järvinen. The death and rebirth of privacy-preserving WiFi fingerprint localization with Paillier encryption. In *IEEE Conference on Computer Communications (INFOCOM)*, pages 1223–1231. IEEE, 2018.
- [106] A Yao. How to generate and share secrets. IEEE Symposium on Foundations of Computer Science, 1986.
- [107] Xun Yi, Russell Paulet, Elisa Bertino, and Vijay Varadharajan. Practical k nearest neighbor queries with location privacy. In *2014 IEEE 30th International Conference on Data Engineering*, pages 640–651. IEEE, 2014.
- [108] Xun Yi, Russell Paulet, Elisa Bertino, Vijay Varadharajan, et al. Practical approximate k nearest neighbor queries with location and query privacy. *IEEE Trans. Knowl. Data Eng.*, 28(6):1546–1559, 2016.
- [109] Xixun Yu, Zheng Yan, and Athanasios V Vasilakos. A survey of verifiable computation. *Mobile Networks and Applications*, 22(3):438–453, 2017.
- [110] Xixun Yu, Zheng Yan, and Rui Zhang. Verifiable outsourced computation over encrypted data. *Information Sciences*, 479:372–385, 2019.
- [111] Dong Yuan, Qi Li, Guoliang Li, Qian Wang, and Kui Ren. PriRadar: A privacy-preserving framework for spatial crowdsourcing. *IEEE transactions on information forensics and security*, 15:299–314, 2019.
- [112] Dongxiang Zhang, Kian-Lee Tan, and Anthony KH Tung. Scalable top-k spatial keyword search. In *Proceedings of the 16th international conference on extending database technology*, pages 359–370, 2013.
- [113] Qi Zhang, Shaojing Fu, Nan Jia, and Ming Xu. A verifiable and dynamic multi-keyword ranked search scheme over encrypted cloud data with accuracy improvement. In *International conference on security and privacy in communication systems*, pages 588–604. Springer, 2018.
- [114] Jingcheng Zhao, Xinru Fu, Zongkai Yang, and Fengtong Xu. Radar-assisted uav detection and identification based on 5G in the Internet of Things. *Wireless Communications and Mobile Computing*, 2019, 2019.

- [115] Qingji Zheng, Shouhuai Xu, and Giuseppe Ateniese. VABKS: verifiable attribute-based keyword search over outsourced encrypted data. In *IEEE conference on computer communications (INFOCOM)*, pages 522–530. IEEE, 2014.
- [116] Yan Zheng, Qian Xinren, Liu Shushu, and Deng Robert, H. Privacy protection in 5G positioning and location-based services based on SGX. *ACM Transactions on Sensor Networks (TOSN)*, 2021.
- [117] Kathryn Zickuhr. Location-based services. *Pew Research*, 679:695, 2013.



ISBN 978-952-64-1088-3 (printed)
ISBN 978-952-64-1089-0 (pdf)
ISSN 1799-4934 (printed)
ISSN 1799-4942 (pdf)

Aalto University
School of Electrical Engineering
Communication and Networking
www.aalto.fi

**BUSINESS +
ECONOMY**

**ART +
DESIGN +
ARCHITECTURE**

**SCIENCE +
TECHNOLOGY**

CROSSOVER

**DOCTORAL
THESES**